The Optimisation of a High Speed Servomechanism

Rayner, Robert

Award date:
2010

Awarding institution:
University of Bath

THE OPTIMISATION OF A HIGH SPEED SERVOMECHANISM

Submitted by Robert M. C. Rayner
for the degree of
Doctor of Philosophy
of the University of Bath
2010

The Optimisation of A High Speed Servomechanism

Robert M. C. Rayner

May 2010

Contents

Abstract

The problem of mechanism design is one of the fundamental, classical engineering problems. The ability to create mechanical devices capable of performing specific tasks and to follow desired motion paths, lies at the heart of the discipline. Numerous different methods of addressing this problem have been developed over the years. The literature shows that extensive work has been carried out over the years to develop methods of optimising mechanism kinematics, generating mechanism designs which can accurately follow given output paths.

When a mechanism is actuated at high speeds, a range of problems are encountered as physical imbalances in the mechanism design are excited, leading to the induction of high frequency harmonic content in its output motion. This harmonic content can greatly limit the maximum operating speed of a mechanism, resulting in loss of motional accuracy and even physical failure of the mechanism. The vast majority of mechanism synthesis methods consider only mechanism kinematics with mechanism dynamics rarely given much consideration, if any at all. The literature indicated that a link exists between the amount of harmonic content present in the output motion of a mechanism and the peak-to-peak magnitude of the variation in drive torque needed to generate that motion.

A prototype servo-mechanical test rig was provided by the project's industrial sponsor company, ITCM Ltd. The test rig possessed a multi-link mechanism, nicknamed the Woodpecker mechanism, which had been observed to have an output motion rich in

harmonic content. This mechanism was modelled and used as a test case for development work. A reconfigurable test rig was also designed and built. Using this test rig, it was possible to assemble and actuate a wide variety of mechanism designs without refabricating parts.

The work described in this thesis details the development of two contrasting approaches of optimising the performance of servomechanisms for operation at high speeds. The output motions of these mechanisms will possess low levels of harmonic content.

The first method was entitled the *Cam Function Generation Method*. This method is applicable for use with existing servomechanisms. Using this method, a variable velocity demand signal was generated. This demand signal takes into account and compensates for the dynamic characteristics of the system. The resultant motion was simulated to yield reductions in peak-to-peak magnitude of drive torque magnitude as well as a reduction in the amplitude of the induced harmonic content the output motion. The method was applied to the prototype test rig. The experimental results confirmed the simulated results.

The second approach tackles the problem of synthesising new mechanism designs taking into account mechanism dynamics and kinematic to generate mechanisms which possess not only good motional accuracy but also good high speed operating characteristics. A mechanism synthesis tool called SWORDS was used to perform the kinematic synthesis whilst a dynamic simulation program entitled DYSIM was used to perform the dynamic analysis using inverse dynamics. The process was applied to the Woodpecker mechanism. A series of three superior alternative mechanism designs were synthesised and a hierarchy of effectiveness between them was identified.

Dynamic models of the reconfigurable test rig, configured in the form of the seed mechanism and the three alternative mechanism designs were created. Inverse dynamic analysis revealed a change in performance hierarchy compared to that identified during the synthesis process. The seed mechanism and the alternative mechanisms were assembled and actuated using a velocity step demand signal to analyse their dynamic

qualities. The experimental results verified the simulation results with the same hierarchy of performance being observed. The Cam Function Generation Method was applied to the seed mechanism and the dynamically most superior alternative mechanism design. It was possible to observe that the alternative mechanisms possessed considerably better dynamic characteristics than the seed mechanism. The Cam Function Generation method was applied to the most superior alternative mechanism design. Additional dynamic improvements were observed.

Acknowledgments

Writing this thesis and performing this project has been a labour of love, involving at times sweat and tears. Without the unerring support of my family, friends and work colleagues, I feel sure that performing this project would not have been as fulfilling as it has proved to be.

In particular I would like to thank my parents and my sister for their amazing love and support. Their words of inspiration have helped to motivate me and lift me up when times were hard. I would like to thank my friends for their amazing friendship and companionship. I would especially like to thank them for providing me with a constant stream of *distractions* to help me relax and take my mind off work. I would particularly like to thank Rosie Harris for kindly volunteering to proof-read the entire thesis!

I would also like to thank the project sponsor company, ITCM Ltd, for their professional and financial assistance. I would particularly like to thank Graham Shirley for his exceptional practical expertise and guidance.

Thanks also go to the EPSRC and the University of Bath for funding the project and to the University of Bath Department of Mechanical Engineering for providing such ideal facilities for me to work in. Last and by no means least, I would like to thank my Project Supervisor, Necip Sahinkaya for all his support and guidance over the course of the project and for keeping my work pointing in the right direction at all times.

Nomenclature

Variables

$\mathbf{0}$ = Matrix of zeroes

$\mathbf{A}$ = ZxZ matrix of $a_{i,j}$ functions

$\mathbf{B}$ = Control action specifying coefficient matrix

$\mathbf{C}_1$ = Matrix of functions $\mathbf{q}$ and t

$\mathbf{C}_2$ = Matrix of function $\mathbf{q}$, $\dot{\mathbf{q}}$ and t

$\mathbf{D}_1$, $\mathbf{D}_2$ = Vectors of function $\mathbf{q}$, $\dot{\mathbf{q}}$ and t

$\mathbf{F}$ = Constraint Jacobian Matrix

$\mathbf{P_{Cam}}$ = Time dependent demand position array

$\mathbf{Q}$ = Vector of generalised inputs

$\mathbf{q}$ = Vector of generalised coordinates

$\mathbf{U}$ = Control input vector

$\mathbf{y}$ = Desired motion specifying coordinates

$\Delta\mu$ = Co-efficient of friction for fluctuation motion

λ_j = Lagrange multiplier

μ_k = Dynamic coefficient of friction

μ_s = Static coefficient of friction

θ_n = Angular position of linkage n

θ_{out} = Output angular position

ω = Angular velocity

A_{dem} = Demand acceleration

$a_{i,j}$ = Function of $\mathbf{q}$ and time

b = Generic coefficient of friction

b_{comb} = System coefficient of friction

f_j = Constraint equation

I_{out} = Output current

i, j, k = Integers

J = Generic inertia

J_{0n} = Inertia scaling factor of linkage n

J_{drive} = Inertia of servo motor, transmission and crank

$J_{driveNoCrank}$ = Inertia of servo motor, transmission and crank mounting components

J_n = Inertia of linkage n

K = Number of degrees of freedom of the required motion

K_{filter} = Integral anti-windup filter gain

Ki = Integral controller gain

Ki_{pos} = Position control loop integral gain

Ki_{vel} = Velocity control loop integral gain

K_m = Controller system gain

Kp = Proportional controller gain

Kp_{pos} = Position control loop proportional gain

Kp_{vel} = Velocity control loop proportional gain

K_{tq} = Servo motor current to output torque proportionality constant

L = Lagrangian function

l_n = Length of linkage n

M = Number of degrees of freedom

m_{0n} = Mass scaling faction for linkage n

m_n = Mass of linkage n

$maxAccel$ = Maximum permissible output acceleration

$maxDecel$ = Maximum permissible output deceleration

$maxVolt$ = Maximum permissible output voltage

N = Gear ratio

n = Linkage number reference

$n_{cycleCam}$ = Number of cycles of crank performed during cam function motion

$n_{cycleWindupReal}$ = Experimentally performed number of cycles of crank performed during cam function motion

$n_{cycleWindup}$ = Number of complete cycles of crank performed during windup

P_{dem} = Demand position

P_{demLog} = Experimentally logged position demand

P_{demSim} = Simulated position demand

P_{out} = Output position

P_{outLog} = Experimentally logged output position

P_{outSim} = Simulated output position

q_i = Generalised coordinate

Q_i = Generalised output

R_1, R_2 = Cost value for sections 1 and 2 of cost function respectively

R_D = Dynamic cost value

R_K = Kinematic cost value

R_{Kin} = Kinematic cost value

RR = Voltage to current conversion factor

R_T = Combined total cost value

R_{Tq} = Torque cost value

R_{Vel} = Velocity cost value

s = Laplace operator

T = Generic torque

$T_{friction}$ = Effective friction induced torque

T_{out} = Output drive torque

T_{outLog} = Experimentally logged output drive torque

T_{outSim} = Simulated output drive torque

t = Time

t_{windup} = Duration of windup period

$V_{camInit}$ = Initial cam function velocity

V_{dem} = Demand velocity

V_{demInt} = Controller internal velocity demand signal

$V_{desired}$ = Desired output velocity

V_{error} = Velocity error

V_{out} = Output velocity

V_{outLog} = Experimentally logged output velocity

V_{outSim} = Simulated output velocity

V_{windup} = System windup velocity demand

$V_{windupEnd}$ = Final velocity at end of windup motion

$V_{windupReal}$ = Experimentally derived windup velocity demand

W_1, W_2 = Cost function weighting factors

Z = Number of generalised coordinates

Units

$degs$ = Angle in degrees

Hz = Frequency in Hertz

kg = Mass in kilograms

m = Length in meters

mm = Length in millimeters

N = Force in Newtons

rpm = Angular velocity in revolutions per minute

rad = Angle in radians

s = Time in seconds

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

The problem of mechanism design is one of the fundamental, classical engineering problems. The ability to create mechanical devices capable of performing specific tasks and to follow desired motion paths, lies at the heart of the discipline. Numerous different methods of addressing this problem have been developed over the years.

In today's modern industrial world, there is an ever increasing demand for machinery capable of operating at high speeds. As operational speeds increase, new mechanism design problems are encountered as dynamic design characteristics become ever more prevalent in the behaviour of the mechanism. Physical imbalances inherently present in the design of a mechanism excite are excited during motion inducing harmonic content in the output motion of the mechanism. At low speeds, the amplitude of this harmonic content is small and thus has minimal adverse effect on the behaviour of the mechanism. The amplitude of this harmonic content increases with actuation speed and at high speeds can reach sufficiently large amplitude to significantly hinder the performance of a mechanism. Common problems experienced by high speed machinery include the loss of motional accuracy, excessive stressing of mechanical components and

the induction of resonance in neighbouring machinery. These problems ultimately lead to limitations being placed on the maximum operating speed of the mechanism.

The problem of mechanism design is a topic which has been the subject of large amount of research in the past with numerous tools being developed to help solve the problem. The vast majority of mechanism design methods developed to date tackle the problem of mechanism design purely from a kinematic point of view and completely neglect the dynamics of the system. The result can be a mechanism design which can follow a specified output path with great accuracy when operated slowly is inaccurate with poor dynamic behavioural characteristics when operated at higher speeds. The work detailed in this thesis seeks to tackle this problem by devising methods of synthesising mechanism designs capable of operating at high speeds without sacrificing accuracy nor operating stability. The methods developed will take into account both mechanism dynamics and kinematics in a unified manner.

1.2 Literature Review

As previously discussed the complex problem of mechanism synthesis is one which has been widely researched over the years. Many different mechanism synthesis procedures have been developed, each with their own individual strengths, weaknesses and characteristics. Solving mechanism synthesis problems can be numerically and computationally very demanding due to the vast number of parameters which need to be handled. Thus mechanism synthesis can be a time consuming problem to solve. Some algorithms have been developed to try to accelerate this process but this is sometimes achieved at the expense of absolute accuracy. Conversely other algorithms have been developed, which aim to achieve highly accurate numerical solutions but at the expense of computational speed.

1.2.1 Kinematic Mechanism Synthesis

Jiménez, Álverez et. Al. [2] break the process of mechanism design into three discrete synthesis tasks all of which must be performed to completely define a mechanism.

1. **Type Synthesis** - This synthesis task requires the designer to define the *type* or design of the bodies which are to make up the mechanism. This includes the design of mechanism links and mechanism joints. The *type* of construction units - such as joints, cams, gears, motors and so on - are also selected.

2. **Number Synthesis** - The second category of synthesis task demands the designer to select the number of each component defined during *Type Synthesis* to satisfy the mechanism design demands. The designer must attempt to choose a mechanism style which best satisfies the purpose of the mechanism. Typical mechanism styles, which a design could employ, could include four bar mechanisms, crank rocker mechanisms and so on. Sources such as mechanism design catalogues [3] can be used to aid design at this stage. In this way, the *number* of mechanism links, joints and so on are defined.

3. **Dimension Synthesis** - Having chosen the geometry of the mechanism through Number Synthesis, the next step is to identify the physical *dimensional* parameters of the mechanism to achieve good kinematic behaviour. The most important parameters to identify are clearly the lengths of the mechanism links and the location of the mechanism ground points. Other parameters such as link thicknesses, link lengths, grounding points, motor locations and so on must also be identified. Jiménez, Álverez et. Al. [2] state that the problems faced by dimension synthesis can be broken into three categories:

   (a) **Function Generation** - Through this form of synthesis, a mechanism is optimised to fulfil a specific discrete *function* or task. Examples of such functions include positioning the mechanism end-effector at a particular point at a specific crank angle or ensuring that a particular part of the mechanism

is at a particular angle when the crank is at a particular angle. A further common aim of this type of synthesis is to ensure that the swept area of the mechanism does not exceed a particular value in order to ensure that the mechanism can operate in a limited operating space.

(b) **Path Generation** - This form of synthesis aims to generate a mechanism, the end-effector of which can follow a desired output path with accuracy, passing through a series of predefined precision points in order.

(c) **Rigid Body Guidance** - In this form of synthesis the mechanism is optimised such that elements of the mechanism body, not just the mechanism end-effector, pass through or around a series of precision points over the course of cycle of crank. A possible reason for such a task might be to ensure that the mechanism does not collide with an obstacle.

Of the three synthesis tasks described, the number of solutions that exist for both type and number synthesis problems can be assumed to be limited to a finite number of discrete combinations. For example, in the case of type synthesis, there may be limit to range and type of motors which are available either due to supply reasons or simply due to financial cost. Similarly in the case of number synthesis, the scope for solutions may be constrained by factors such as financial cost and the availability of parts and materials.

Performing experimentation in the areas of type and number synthesis has been shown to be potentially advantageous, with significant gains in performance sometimes achievable. For example Kirecci and Dulger [4, 5] performed a detailed study of a *hybrid* system driven by two servo motors, one driving at a constant speed and the other driving with a variable velocity profile. They compared the performance of this system to an equivalent system driven by two servo motors both driven at a constant speed. They concluded that for a given cyclic rate, the hybrid system would consume less energy per cycle. Yuan, Gilmartin and Douglas [6] quantify this energy saving to potentially be as much as 70%. A common problem faced when driving a system with multiple motors is the problem of motor synchronisation, with any deviation causing significant

motional errors. Robust motor control strategies must be used in such systems. Clearly the problem of synchronisation is not encountered by mechanisms driven by a single servo motor.

Due to the discrete and situational nature of type and number synthesis, these stages of analysis are difficult to automate. The use of experience, intuition and detailed knowledge of the available components on the part of the designer is therefore needed to optimise these stages of synthesis. Resources exist to help designers match the desired output path with the number of links and the number and types of joints to include in the final mechanism, as well as with the pattern of link interconnection. One example of such a resource is the *Mini-Atlas of Linkage Design* by Fichter, Wagner et. al [3].

Unlike the discrete parameters dealt with in number and type synthesis, physical dimensions are continuous variables. Hence in the case of dimension synthesis, there are an infinite number of possible configurations which the mechanism could assume. Some configurations may be unfeasible and impossible to assemble, whilst others may be feasible but possess other undesirable characteristics, such as only being able to achieve a limited number of configurations or *poses* within a certain range of crank angles. The poses achieved by the mechanism at these limiting crank angles of operation are known as *limit* and *dead* configurations [7]. These poses represent limits of operation as to pass through them requires infinite amounts of propulsive torque to be exerted on the mechanism crank by the drive motor. Other solutions may be feasible, but lack the ability to follow the desired output path. With so many possible configurations, much work has been carried out to derive methods to analyse and quantify the quality of a mechanism configuration and thus identify an optimal solution.

For any given mechanism configuration, further numerical expressions can be written to analyse specific numerically quantifiable aspects of the mechanism design and to quantify the quality of a design. Examples of characteristics to be interrogated include mechanism mobility [8–10] mechanism swept area [8] or most commonly deviation of

end-effector from the desired path [2,6,8,10–21]. Using quality quantifying expressions, called *cost functions*, quality indicating characteristics are compared with an ideal case. By iterating using different mechanism parameter combinations it is possible to analyse a variety of different mechanism designs each with differing levels of quality. The better the design and the closer the quality indicator is to the ideal value, the lower the value of *cost*. Using this approach, the problem of mechanism synthesis is reformulated to take the form of multi-parametric optimisation problem. To solve this problem, an optimisation method must iterate through different parameter value combinations and identify the solution with the lowest overall cost. The biggest issue with addressing the problem in this way is that the solution space, which the optimisation method must search is infinitely vast. Typically optimisation techniques require initial parameter values to be provided to initialise the method. Doing this positions the technique in a particular *neighbourhood* of the solution space. In this neighbourhood, a region of minimum cost will exist which a search method may identify. Care must be taken since although this identified solution is a *local* minimum for the neighbourhood, other neighbourhoods may possess minima with even lower cost values. The single elite minimum with the lowest cost value of the entire solution space is referred to as the *global* minimum.

Traditional parametric optimisation techniques such as the Newton-Raphson method or Euler's method, do not possess the ability to search multiple solution neighbourhoods and are easily fixated by local minima [20]. Identifying global minima is a difficult task since the behaviour of the cost function is not known in space [14]. To overcome this issue, a lot of work has been carried out to create new optimisation techniques capable of searching multiple neighbourhoods, to increase the likelihood of identifying the solution corresponding to a global minimum cost value.

A wide variety of approaches are suggested in the literature. The most common include the use of population based, stochastic, evolutionary numerical methods such as use of genetic algorithms (GA's) [8, 12, 15, 19, 22]. Price and Storn developed a GA style method known as Differential Evolution [23]. Liu and Xiao developed a method based

on biological autoimmune systems method known as Artificial Immune Searching (AIS) [16] whilst Smaili and Diab developed an optimisation method inspired by the principle of pheromone deposition by ants [24].

New numerical optimisation techniques have been developed to search a solution space. These include the Tabu search by Smaili et. al. [20], the Chaotic Descent method [14] by Jovanovic and Kazerounian and the Geometric Centroid of Precision Positions technique by Shiakola, Koladigy and Keble [25]. Hansen developed a method of automating number and dimension synthesis using his Time Varying Dimensions method [13].

A common method used in optimisation methods to navigate the solution space is to analyse it in terms of its gradient. In this way, a method is capable of detecting the proximity of minima by looking for turning points in the solution space identifiable by regions of zero gradient. Analysing the variation of the gradient of the solution space allows the method to generate a profile of it providing clues to the location of minima. A good example of such a gradient based method is the Tabu-Gradient Search by Smaili [20] but numerous others exist [9, 18, 25]. One of the difficulties of tackling the problem in terms of gradients is the fact that the calculation of gradients can be difficult and computationally expensive to do. Accurate calculations of gradients are essential to ensure meaningful results are obtained from such methods [11]. Mariappan and Krishnamurty described the nature of this problem and addressed it by proposing a matrix based method of calculating exact gradients [26]. Avilés et. al. [11] addressed this problem in a different way using error gradient functions. In this paper they also described an optimisation method which optimises a cost function using second derivative information. The method uses the Newton-Raphson method to perform the parameter optimisation.

It has been shown that dynamically changing the boundaries of search neighbourhood over the course of the optimisation can help to greatly improve the accuracy and/or speed of a search method. Some optimisation methods possess the capability to perform such operations alongside performing a numerical optimisation method. A commonly

used technique is Fuzzy Logic (FL). Laribi, Mlika et. al tried using FL in conjunction with a classical GA technique to guide the evolutionary process [15]. They found that the use of FL did not accelerate the optimisation process but improved its accuracy. Yuan, Gilmartin and Douglas also use a FL based technique to optimise their mechanism designs [6, 27]. In their method, each design variable forms a fuzzy set. Membership functions define boundaries for each optimisation objective. Jovanovic and Kazerounian use the principle of Julia sets to divide up the solution space in their Chaotic Descent method [14].

Zhang et. al. [28] looked at the problem of mechanism synthesis in a different way, developing a method of synthesising conceptual hybrid mechanisms using State Space theory [29]. Leal and Dai visualised the problem in terms of origami paper folds and developed a new mechanism class [30].

1.2.2 Methods of Optimisation for the Reduction of Harmonic Content

The vast majority of the synthesis methods discussed thus far are all designed to tackle the *dimension synthesis* problem. The primary aim of these methods is to generate mechanism designs, which can follow a desired output path with accuracy. Often the problems of function generation or rigid body guidance are not dealt with.

A specific aspect of function generation, which is of particular interest to this investigation, is the behaviour of a mechanism when it is actuated at high speeds. When actuated at high speeds, physical imbalances in the mechanism are excited, leading to the induction of harmonic content in the output motion of the mechanism. The amplitude of this harmonic content increases with actuation speed.

Yuan and Rastegar [27, 31] describe methods of redesigning existing mechanisms to reduce the amount of harmonic content present in their output motions. They explain that the harmonic content induced in the output motion of a mechanism is caused by

non-linearities caused by closed loop chains and rotary joints during motion. They also believe that flexibility in the joints of the mechanism may also play a role in vibration induction. The energy possessed by these vibrations increases with operating speed. Should an induced frequency be induced that matches that of another component in the system or indeed in another machine in the vicinity then these components will resonate, leading to the induction of further harmonic content.

Physical vibrations caused by this harmonic content can have highly detrimental effects on system performance, particularly if the amplitude of these vibrations are large. Motional accuracy can be adversely affected, since the nature of the vibrations in the output motion are unpredictable. The presence of harmonic content also increases the energy consumption of a system, since the motor not only has to do work to drive the mechanism along the desired trajectory but also to power the harmonic content. The harmonic content also increases the magnitude of the forces being transmitted within and between the mechanism links and connection points thus putting extra stress on these components hence decreasing their mechanical life. A common method of limiting the severity of these problems is simply, to limit the magnitude of the harmonic content that can occur in mechanism by limiting the maximum operating speed of the system even if a higher operating speed is more desirable.

Yuan and Rastegar [27, 31] implied a direct link between harmonic content in the output motion and the peak-to-peak magnitude variation of motor actuation torque, which the servo motor needs to exert on the mechanism over a complete cycle. They stated that reducing the amplitude of the harmonic content in the output motion of the mechanism can be achieved by reducing the peak-to-peak magnitude of the cyclic actuation torque variation needed to generate that motion. They also suggested two methods of modifying a mechanism to reduce the amplitude of the harmonic content in the output motion of a mechanism. They suggested that using cams [27] or smart materials such as piezoelectric stacks [31] could be used to dynamically vary the length of the link lengths in order to achieve this effect. Methods of designing the cam and controlling the smart materials are suggested. Smart materials are said to be effective at

minimising small amplitude, high frequency vibrations, whilst cams are more effective at compensating for lower frequency but larger amplitude vibrations. They stated that both cams and smart materials could be used in parallel to create a mechanism with good dynamic qualities. A detailed investigation of the electro-mechanical properties, possible applications and methods of implementation of piezoelectric stacks was carried out by Wang, Ehlers and Neitzel [32]. Much work has also been documented regarding the reduction of harmonic content in the output motion of a mechanism through the use of an optimum balancing method [33–35]. Fundamentally, this method involves the addition of masses to the mechanism links to reduce the magnitude of the out of balance forces in the system. These methods whilst they may well be effective can be potentially complex and costly to implement require large amounts of development work to achieve. Resultant mechanisms may be highly complex. It may also be undesirable to add extra components to a given mechanism due to mechanism access or space restrictions.

Connor et. al. [8], Kochev [33] and Conte et al. [36] tackled the problem of harmonic content in a more direct manner, using a cost-based optimisation method. Given an initial mechanism and an ideal output motion path, the mechanism is re-synthesised using a more complex cost function, which in addition to the typical path following expressions also possesses an expression, which penalises higher order (2nd harmonic and above) frequencies in the output motion of the mechanism. The fundamental frequency used in this analysis is the cyclic frequency of the mechanism. For each candidate solution, Fourier analysis is used to determine the harmonic content present in the resultant output motion. A hill climbing optimisation algorithm is used. Yuan, Gilmartin and Douglas [21] described a similar method but as previously discussed use a Fuzzy theory based optimisation method.

Analysing the harmonic content of a motion also provides a convenient way of classifying and identifying a particular style of mechanism [8]. Nie and Krovi [17] describe the application of a mechanism synthesis program, which uses Fourier transforms to perform both dimensional and functional synthesis.

It has been identified that mechanisms possess *critical* operating speeds at which vibrations in the output motion excite resonant frequencies of components within the mechanism itself, resulting in sudden increases in harmonic content amplitude as actuation speed approaches these critical values. Yu and Cleghorn [37] devised a method of defining these *critical* running speeds using finite element (FE) analysis together with Lagrangian analysis such that these critical actuation speeds can be avoided. Cleghorn, Tabarrok and Fenton devised a method of obtaining FE equations for use in mechanism stability analysis using truncated Fourier series [38]. Cleghorn, Tabarrok and Fenton also devised an alternative method of identifying stable operating speed ranges using the solution of the Mathieu equation to define stability boundaries [39].

Yu and Smith [40] analysed the influence of physical cross-sectional parameters of a link on its resonant frequency. They concluded that varying the width of a link changes its resonant frequency considerably, whereas varying its height did not. This work compliments work carried out by Cleghorn, Fenton and Tabarrok, which describes a method of defining the minimum cross-section of mechanism links with respect to physical, inter-componental loading [41].

1.2.3 Output Motion Path Representation and Accuracy Comparison

When performing geometric synthesis, good candidate solutions must be able to follow a desired output path with accuracy. Typically, the continuous desired path is discretised to generate a series of discrete *precision points*, to which the mechanism end-effector should pass through or close to. The ability to compare this desired path with the output path of a candidate mechanism is a process which is implied as being easily achievable in all of the literature encountered so far in which geometric synthesis is performed. Rarely however, do these papers describe *how* this operation is performed. The problem of path comparison is highly comparable to the problems of shape representation and similarity matching. Such problems have been the basis of much research with potential applications in bio-medical engineering and image pro-

cessing, as well as others. Antani et. al. [42] highlight the presence of two main styles of approaching the shape representation and similarity comparison problems.

The first style uses a method known as *shape deformation*. This approach to the problem can be visualised by first considering the candidate shape being scrutinised as an elastic material. This shape can be deformed (stretched or compressed) until it assumes the form of the target shape. Work must be done to deform the candidate shape. The better the fit between the two shapes, the lower the amount of work that must be done to deform the candidate shape. Quantifying the amount of work needed for the candidate shape to assume the form of the target shape generates a numerical measure of the similarity between the two shapes. Clearly, the lower the amount of deformation work necessary, the better the level of fit between the two shapes. Alternative methods of formulating the deformation are also detailed by Antani et. al. Such methods include Gradient Vector Flow (GVF) and Eigenshape decomposition.

The second style of analysis is more abstract and involves representing the target shapes as implicit polynomials. Affine transformations are often used to interpret the shapes, described in terms of Cartesian coordinates, to allow the shapes to be interpreted more easily in an alternative domain. Examples of such transforms include the use of wavelets as done by Alferez and Wang [43] and the use of Fourier transforms as done by Mullineux and Yuan, Gilmartin and Douglas [21] as well as discrete cosine transforms [44]. The shape can then be compared through the comparison of the resultant polynomials using non-graphical numerical methods.

An alternative method of interpreting shapes is to represent the shape in question as a Bernstein basis curve (also known as Bezier curve) and in particular as a B-spline as described by Farin [45] and by Rogers and Fog [46]. In this way, the shape can be formulated as a single parametric function. The problem with using Bernstein basis curves is that these functions can become of a high order, particularly if the shape is complex, making it difficult to manipulate. This problem can be avoided by representing the shape as B-spline. A B-spline is fundamentally a series of Bernstein

basis curves joined together using a *knotting function*. Using B-splines, a complex shape can be described as a stable, low order function. The representation of the desired path, in terms of precision points, is particularly useful as these points can be used to position the control track for the curve. Manipulation of the control track of the candidate shape has implications for shape comparison using graphical shape deformation type methods. Conversely, the representation of low order polynomials also has implications for shape comparison using numerical methods.

When comparing two shapes, three main issues exist that all good shape comparing algorithms should be able to recognise and deal with. Not addressing these problems may lead to incorrect and unfair comparisons being made pro-longing the search.

The first issue is that of shape scaling. When this issue occurs a shape may be generated that is geometrically very close to the target shape but scaled either up or down by a constant factor. In the case of geometric synthesis for mechanism design, not identifying this condition could lead to the synthesis process being prolonged when a quick solution may be achieved simply by scaling the mechanism link dimensions.

The other two issues are those of shape translation and rotation. A candidate shape may be geometrically very similar to the target shape, but either rotated and/or translated about a point in space. In the case of geometric synthesis for mechanism design, the synthesis process may again be prolonged when a simple rotation and/or translation of the mechanism design may yield a solution.

1.2.4 Existing Computer Based Synthesis Tools

Of the mechanism synthesis and analysis methods previously discussed, virtually all of the procedures have been automated and performed using a computer. Most are computationally intensive. Some of the methods have not progressed further than prototype stand alone methods. Others however have been developed further and incorporated as sub-components into larger scale software packages alongside other

supplementary methods and subroutines. An example of such a software package is the synthesis tool developed by Nie and Krovi [17].

For the purposes of this investigation, it is envisaged that two programs, both developed by academics within the University of Bath, will be used to perform the required mechanism synthesis and analysis tasks. These programs are entitled SWORDS, developed by G Mullineux and DYSIM, developed by MN Sahinkaya. SWORDS is a software package possessing powerful mechanism synthesis and multi-parametric search capabilities, whilst DYSIM is a multi-body and multi-physics modelling, simulation and analysis tool and as such complement each other well.

1. **SWORDS** is a software package developed by G Mullineux at the University of Bath. The program, in its most fundamental guise is a numerical optimisation and constraint modelling program. Using SWORDS a user can define any number of numerical constraints. After defining initial conditions, the program will attempt to optimise the numerical variables to attempt to achieve an optimum parameter combination. Optimisation is performed using a cost based optimisation technique known as the Powell's method [48].

   Where suitable, a graphical representation of the problem represented by the constraint equations can be designed by the user and displayed on the screen. SWORDS can update this graphical display, showing the user not only the final, optimised configuration, but also the intermediate, candidate solutions. SWORDS uses a geometric modeller known as *ACIS* [49] as a graphics engine to display its images.

   SWORDS is of particular interest to this investigation due to its powerful mechanism synthesis capabilities. Before SWORDS can perform geometric synthesis, the user must provide the program with some information. They must specify precision points in space to define the desired output motion of the mechanism. *Type* and *number* synthesis must be performed manually by the user. To aid with these tasks, SWORDS possesses its own internal mechanism catalogue sim-

ilar to Fichter's Mini-Atlas of Mechanism Design [3]. If desired, the user can request SWORDS to search the catalogue to identify a suitable mechanism style. They must also specify initial values for parameters, such as link lengths and/or grounding points. This stage of the process corresponds to the *type* and *number* stages of mechanism synthesis discussed in Sec.1.2.1.

With this information, SWORDS can now assemble the multi-parametric constraint equations for the problem and perform *number* synthesis. It traverses the solution space, testing different combinations of parameter values. Each candidate solution is tested for path tracking ability against the desired path and a cost value derived. The synthesis algorithm attempts to be sympathetic to the influences of scaling, rotation and translation of the resultant path.

Resultant solutions are ranked according to cost. The user is able to view these solutions. The user is also able to animate the mechanisms and view the progression of the candidate solution over a complete cycle of crank. The resultant output path can be displayed on the screen alongside the desired path for additional ease of comparison. On occasions, it may be desirable not to use the solution with lowest cost due to functional reasons such as mechanism size or exceptional accuracy in particular portions of the motion. The geometric parameter values for any of the solutions can be exported to an ASCII file for analysis using other tools.

It is important to note that the SWORDS is only capable of performing kinematic optimisation. Resultant mechanisms, whilst kinematically very accurate, may possess undesirable dynamic characteristics. For example, the mechanism may be highly imbalanced, leading to the induction of large amounts of harmonic content as actuation speed is increased. SWORDS does not possess the capability to analyse mechanism dynamics. To perform this task a second program entitled *DYSIM* should be used.

2. **DYSIM** is a versatile and powerful multi-body dynamic analysis tool developed by Dr M N Sahinkaya. Using this tool, models of even very complex multi-body systems can be created with ease. The program uses the principles of Lagrangian

dynamics to model the system. This approach considers each body of the system in terms of the amounts of kinetic and potential energy it possesses. Using Lagrange's Formula [50], the equations of motion of the complete system can be automatically derived. For the purposes of mechanism design, each mechanism link is considered to be an individual body. Further energy functions can be entered to the system to simulate effects such energy losses in the system through phenomena such as friction.

Once the equations of motion have been derived, the user must define the dependent and independent system parameters. The user must also specify initial conditions for the independent variables. These initial conditions can be expressed in terms of simple units (i.e. displacements or angles) and first derivatives (i.e. velocities). Using this information, DYSIM calculates, using the equations of motion, corresponding initial conditions for the other dependent variables. On some occasions, more than one initial configuration may be possible. For example, it may be possible to assemble a mechanism in two ways. DYSIM may not identify the desired configuration. To ensure that the desired configuration is identified by DYSIM, it may be necessary to specify further, approximate values for the initial conditions of some or all of the dependent variables, to effectively point the program towards the desired configuration.

With the creation of the model now complete, the model can be exported from DYSIM in the form of a function for use within a MATLAB/Simulink [51] model, allowing complex analysis of the system to be carried out. Using DYSIM, two forms of Simulink block can be created, enabling the system to be analysed in two different ways. The first method is *forward dynamics*. Through this method, the user must specify the generalised input signals to the system. Using the equations of motion of the system, the response of the system to the input signal is calculated.

The second method is called *inverse dynamics*. Through this procedure, the desired output of the system is specified by the user. Using the equations of motion of the system, the ideal input signal needed to generate this output motion

is calculated. Details of the methods used by DYSIM to perform inverse dynamic calculations can be found in [52].

1.3 Aims & Objectives

The overall aim of this project is to develop methods and techniques to generate mechanisms capable of operating at high speeds with good dynamic and kinematic characteristics. The aims of the project can be broken down into two main parts:

- Work will be done to develop control strategies for existing, pre-constructed mechanisms, to improve their ability to operate at high cyclic rates. These strategies will seek to improve high speed mechanism performance without the need for any physical modifications to be made.
- Work will be done to investigate methods of synthesising new mechanisms, which are optimised both in terms of kinematic accuracy as well as dynamic quality.

In both cases, theoretical simulation work will initially be carried to develop and analyse the effectiveness of potential methods. Effective methods will then be validated experimentally under laboratory conditions. If need be, custom laboratory equipment will be designed and manufactured.

1.4 Layout of The Thesis

The work discussed in this document describes the development of two methods of improving the performance of high speed servomechanisms, with the primary aim of enabling them to operate at a given speed with reduced levels of harmonic content in their output motions.

As a basis for the work, a prototype mechanism and industrial test rig was supplied by the sponsor company of this project, ITCM Ltd. The assembly and modelling of this test rig is detailed in Ch. 2 of this thesis. In Ch. 3, a performance enhancing method named the *Cam Function Generation Method* is developed using computer simulations and then experimentally applied to the test rig.

In Ch. 4 work is detailed in which a novel multi-stage mechanism synthesis method was developed. To experimentally validate this method, a new, reconfigurable test rig was designed and constructed. The design, construction and modelling work of this test rig is described in Ch. 5. Using this test rig, the two methods previously described in Chapters 3 and 4 were both experimentally validated in Ch. 6.

A summary of the conclusions of the work detailed in this document can be found in Ch. 7.

Chapter 2

The Industrial Prototype

As a basis for experimental and theoretical work, a prototype test rig was provided by the project's industrial sponsor, ITCM. The test rig consisted of a typical multi-link mechanism propelled by a typical industrial servo motor and corresponding motion control hardware. This chapter details practical work carried out to collect data used to create a model of the test rig.

2.1 Hardware Setup

As previously stated, the test rig being investigated consisted of a multi-link mechanism propelled via a servo motor controlled by servo control hardware. The various constituent parts of the test rig will be discussed in turn in this section.

First consider the mechanism, a photograph of which is shown in Fig. 2.1.

A three-dimensional CAD model of the mechanism is shown in Fig. 2.2 whilst a two-dimensional line based drawing of the mechanism is shown in Fig. 2.3. In Figures 2.2 and 2.3, the labelling convention used to identify each of the different elements of the mechanism are shown. Of particular note are the locations of each of the three

Figure 2.1: A photograph of the Woodpecker mechanism installed on the test rig

grounding points, *Ground 1*, *Ground 2* and *Ground 3* via which the mechanism was anchored to the ground. The mechanism was propelled via its crank link, labelled *1*. The crank link rotated around its ground point, *Ground 1*, in an anti-clockwise direction. The mechanism crank was connected to the drive motor via a belt drive transmission system (discussed later in this chapter).

The mechanism was nicknamed *The Woodpecker Mechanism* due to the back and forth rocking nature of its main body, labelled *5* and *7*, during motion.

The system represented by the test rig simulates a section of subsystem of a much larger manufacturing machine. The function of this subsystem was to push small envelopes into a neighbouring hopper, one per cycle. The mechanism had a desired output rate of 600 products per minute thus it had to achieve an average cyclic rate of 600 rpm to satisfy this demand. The envelope pushing portion of this subsystem is simulated by the test rig.

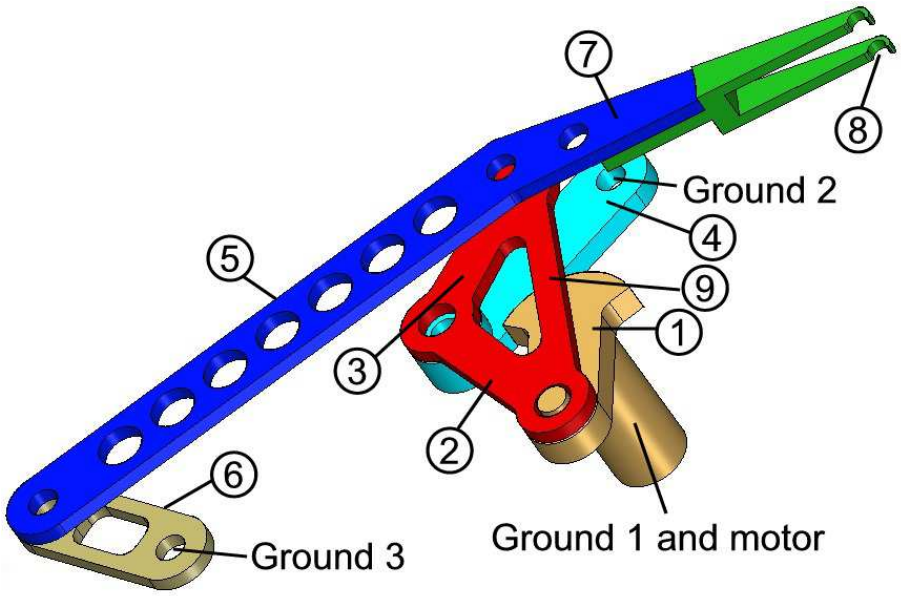


Figure 2.2: A CAD drawing of the Woodpecker mechanism

The end effector of the mechanism is labelled as element *8*. The *D shaped* output path of the mechanism is shown in Fig. 2.3. During the upper, flat portion of the motion, the mechanism interfaces with the product, pushing it into the hopper. The end-effector then turns sharply and recoils, following a curved arc to start a new cycle. The most important characteristic of the output motion is the region in which the orbit approximates to a straight line. This region of the orbit is important since it is in this region of the motion that the mechanism interacts with the product. When designing the mechanism, specific work was carried out to minimise vertical deviation of the end-effector in this region. It is also critical that at the end of the straight portion of the motion, the end-effector should turn away from direction of travel of the straight portion of the motion to avoid a collision between the end-effector and the hopper. The form of the curved recoil portion of the motion was less important, since the mechanism would merely be passing through free space.

The mechanism crank is propelled by an Electro-Craft S-4075 industrial servomotor. The drive shaft of the motor is connected to the mechanism crank via a synchronous, toothed belt transmission system. A motor to crank gear ratio of 3:1 exists. A belt tensioning pulley was used to take up the slack in the belt. The motor and belt drive

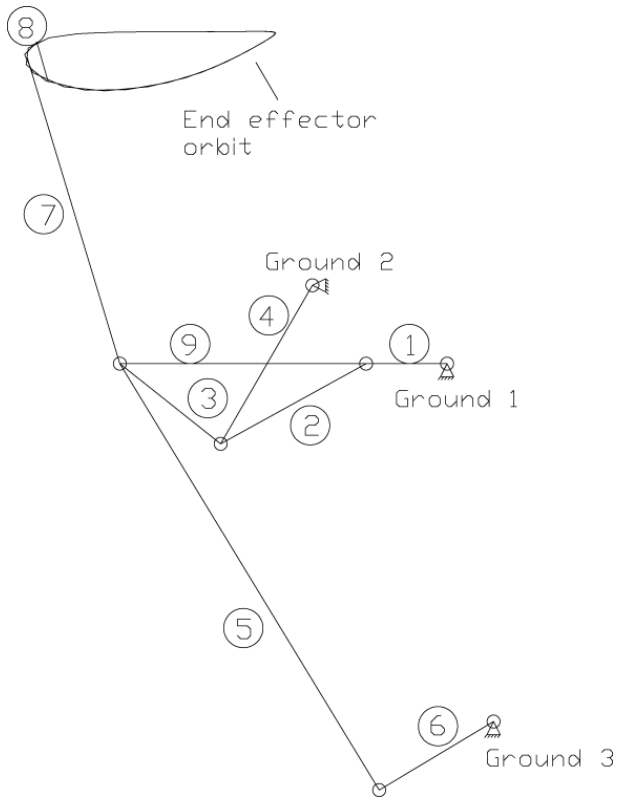


Figure 2.3: A schematic layout of the Woodpecker mechanism

transmission system is depicted in Fig. 2.4

To control the motion of the motor, an external control computer equipped with a Deva004 motion control card is used. The Deva004 is connected to a BRU-DDM30 motor drive unit. The BRU-DDM30 is also connected to the control computer allowing it to be configured and controlled as required using the BRU Master drive control software package [53].

Signals can be passed into and out of the drive unit via a single 22-pin communication port on the BRU-DDM30. The purpose and definition of each of the pins on this connector are detailed in the BRU-DDM30 User's Manual [54]. To allow easy access to each of the individual pins on this port, an external breakout connector was used. A photograph of this breakout board can be seen in Fig. 2.5.

With reference to the BRU-DDM30 Users Manual, it was stated that the BRU unit possesses the capability to output two analogue status signals simultaneously. The

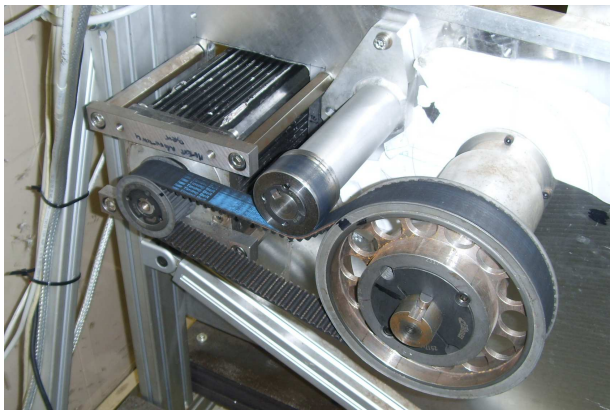


Figure 2.4: A photograph of the servo motor, controller, and belt drive transmission of the Woodpecker mechanism test rig. (The mechanism is mounted on the opposite side of the rig, behind the base plate as depicted.)

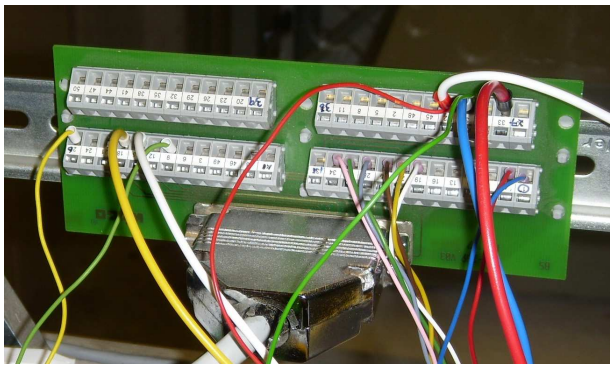


Figure 2.5: The breakout board

definition of these status signals could be configured by the user using BRU Master. These signals are output via two pins on the 22-pin connector. Having identified these pins, a data acquisition device (DAQ), shown in Fig. 2.6, was connected to the relevant pins on the breakout board. The DAQ was in turn connected to a external computer, where the data was logged and stored.

The system was observed to possess a motion which is noisy and rich in harmonic content, making a good candidate for development work.

Figure 2.6: The data acquisition device (DAQ)

2.1.1 The Motor Control Strategy

The motion control of the servomotor was carried out by a control computer equipped with a Deva004 motion control card. The Deva004 was connected to a BRU-DDM30 servo drive via a series of connections made using the breakout board and communication ports on the Deva004 card. The BRU-DDM30 was connected to the servo motor and also connected to the control computer via an RS-232 connection. Suitable software was loaded onto the control computer to enable it to interface with the two devices. In the case of the Deva004, this software took the form of a series of system drivers and Visual Basic scripts [55]. To communicate with the BRU-DDM30, BRU Master was used as previously discussed. Using BRU Master the user is able to pass commands to the drive unit to configure it and also to actuate the motor. Motor motion information, measured using optical encoders installed in the S-4075 servo motor, was fed back to the BRU Master control software via the BRU-DDM30. Using this information, BRU Master was able to calculate motor motion variations such as velocity, position, acceleration and so on. Using BRU Master, it was possible to monitor the variations of these variables but not to log these variations. To log this data the use of

the external Data Acquisition unit (DAQ) was necessary. A summary of this strategy is depicted in Fig. 2.7.

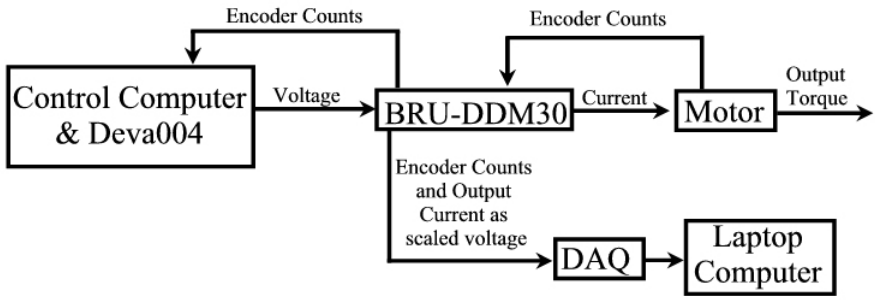


Figure 2.7: An overview of the Woodpecker mechanism test rig control architecture

The BRU-DDM30 can operate in three different modes:

1. Velocity Mode
2. Position Mode
3. Torque Mode

In each of these modes, the BRU-DDM30 unit acts to regulate the motion of the motor with particular focus on the respective operational variable. The behaviour of the drive is governed by user defined parameters. So for example, when operating in *Velocity Mode*, the drive will act to ensure that the motor achieves a user defined *velocity*, accelerating and decelerating at user defined rates. Drive position is not considered. Similarly, when operating in *Position Mode*, the drive will seek to move the motor to a user defined *position*. The motor will accelerate to a maximum speed before decelerating to a standstill at the desired position. Acceleration and deceleration rates, as well as the maximum velocity, are all specified by the user. Position and velocity control regulation is carried out using a traditional PI control strategy.

When operating in *Torque Mode* however, the drive unit monitors neither the speed nor the position of the motor. Instead, the unit monitors the amount of torque being exerted by the motor at its output shaft. The magnitude of the torque demand must

be supplied to the BRU-DDM30 as voltage from an external source, in this case the Deva004. The external signal must be supplied as voltage. The signal is fed to the drive unit via the relevant connection on the breakout board. In this mode, the drive essentially acts as a slave, signal amplifier unit, acting to translate voltage input signals from an external source into output command current signals which are passed to the motor to generate a motion. The magnitude of the motor command current output signal is directly proportional to the magnitude of the input voltage signal. The voltage-to-current proportionality constant is user definable and can be varied using BRU Master. Similarly, the magnitude of the motor output torque is directly proportional to the magnitude of the command current passed to it from the BRU-DDM30. In this case, the constant of proportionality is a physical characteristic constant of the motor itself, K_{tq}. The value of K_{tq} is a pre-calculated physical constant and can be found in the S-4075 servo motor Users Manual [54].

One of the limitations of the BRU Master control software is that it only provides a facility for the user to demand very simple motions, such as constant speed motions or simple point to point motions. Using BRU Master alone, it is not possible to command the motor to perform more complex user defined motions, such as variable velocity motion profiles. To overcome this problem it was necessary to use an external motion control device to regulate the motion of the servo motor. The Deva004 motion control card was identified as a suitable and cost effective device to carry out the task. The card possesses a number of analogue and digital inputs and outputs through which it can monitor motor status signals and output command signals. With the BRU-DDM30 operating in torque mode, the Deva004 can read motion status signals from the BRU-DDM30 and respond with output motion command signals. The Deva004 uses a PI style control strategy for both velocity and position control with a separate loop for position and velocity respectively. Motor position information can be fed directly to the Deva004 from the motor encoders, enabling motor dynamics information to be monitored accurately and quickly.

The Deva004 is controlled using Visual Basic scripts which can call a wide variety of

functions embedded in the Deva004 driver software. These commands perform a variety of tasks, including data processing and output motion scheduling. Using these functions, simple step and ramp style motions can be defined, as well as complex position dependent velocity profile type motions. The definition of controller parameters are also defined using these commands. Motor encoder data is fed back to the Deva004 from the S-4075 servo motor, which the Deva004 uses to monitor and control the motion of the motor. Resultant variables, such as motor position, velocity and acceleration can be logged to an external file by calling suitable functions. Motion control signals are output from the Deva004 as voltages and passed to an external servo motor amplifier, in this case the BRU-DDM30. The Deva004 is only capable of operating in position control mode.

2.2 Modelling the Test Rig

In order to improve the performance of the test rig, it was first necessary to develop a good theoretical model of the system, with the intention of using the model to develop potentially performance enhancing system modifications. If a method proved effective in the simulation, it could then be tested experimentally using the test rig.

When modelling the test rig, the system was considered in three discrete parts, each of which were initially modelled separately:

1. The Mechanism
2. The Controllers (Deva004 and BRU-DDM30)
3. The Transmission

Once all the portions of the system had been modelled, the three models were combined to create a single model of the complete test rig. Modelling and system parameter estimation work was carried using data collected experimentally from the test rig. The test rig was excited using a variety of demand signals and the responses of the system logged. Equivalent demand signals were passed into the model of the test rig and the simulated responses of the system logged. The simulated and experimental responses were then compared. If the two response correlated well then the model was said to be valid.

2.2.1 Modelling the Mechanism

Using data taken from a CAD model of the Woodpecker mechanism, a model of the mechanism was constructed using DYSIM [56]. To construct the model, DYSIM requires that the user specify the mass, inertia and location of the centre of mass for each link, as well as specifying the location of each ground point. The mass, inertia and link

length data of the individual elements of the mechanism as taken from the CAD model are summarised in Tab. 2.1.

Name	Link Number	Length (mm)	Mass (kg)	Inertia (Nmm^2)
Crank	1	62	0.927	901
Connector	2	127	0.310	1420
(Tertiary Link)	3	103		
	9	188.3		
Upper pivot	4	144	0.414	1310
End-effector	8	n/a	negligible	negligible
Spine	5	348.86	0.482	10550
(Tertiary Link)	7	245.19		
Lower pivot	6	102	0.123	290

Table 2.1: The physical data of the Woodpecker mechanism. (Note Links 5 and 7 lie at an angle of 14.44° to one another.)

Note that element 8 of the mechanism represents the end effector of the mechanism. This element is a theoretical point and therefore has no mass or inertia value.

The locations of the ground points of the mechanism are summarised in Tab. 2.2.

Ground Point Number	x coordinate (mm)	y coordinate (mm)
1	0	0
2	60	-121
3	17.5	-274

Table 2.2: The locations of the ground points of the Woodpecker Mechanism

Using this model, it was possible to analyse the mechanism using both forwards and inverse dynamics to gain a full understanding of the behaviour of the system.

DYSIM uses Lagrangian Dynamics to generate the equations of motion of the system. For any system with M degrees of freedom and with $Z \succeq M$ generalised coordinates of motion whereby $q_i, i = 1 \cdots Z$, the Lagrangian equation is:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} + \sum_{j=1}^{Z-M} \lambda_j \frac{\partial f_j}{\partial q_i} = Q_i \quad i = 1 \cdots Z \tag{2.1}$$

The definition of all the variables in Eq. (2.1) can be found in the Nomenclature. The Lagrangian function L is the difference between the kinetic and potential energy

possessed by the system and can be written in a more general form.

$$L = \frac{1}{2}\sum_{i=1}^{Z}\sum_{j=1}^{Z} a_{i,j}\dot{q}_i\dot{q}_j + \sum_{i=1}^{Z} a_{i,0}\dot{q}_i + a_{a,0} \tag{2.2}$$

As superfluous coordinates exist in the system, $Z - M$ constraint equations are needed.

$$f_j = 0, \qquad j = 1 \ldots (Z - M) \tag{2.3}$$

Inserting Eq. (2.2) into Eq. (2.1) and double differentiating Eq. (2.3) results in the following $2Z - M$ algebraic differential equation:

$$\begin{bmatrix} \mathbf{A} & \mathbf{F}^T \\ \mathbf{F} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{D_1} \\ \mathbf{D_2} \end{bmatrix} + \begin{bmatrix} \mathbf{Q} \\ \mathbf{0} \end{bmatrix} \tag{2.4}$$

In Eq. (2.4), $\mathbf{D_1}$ and $\mathbf{D_2}$ are functions of $\mathbf{q}$ and $\dot{\mathbf{q}}$. This can be solved to obtain the $2Z - M$ unknowns, namely the second derivatives of generalised coordinates and Lagrange multipliers. The second derivatives can then be integrated twice to obtain the system forward dynamic response. The time history of Lagrangian multipliers is automatically calculated and can be used to calculate the forces of constraints.

When performing inverse dynamic analysis, the control inputs required to generate a desired output are calculated in terms of second derivatives of the generalised coordinates. Assuming that the required motion has $K = M$ degrees of freedom, it can be represented as follows:

$$\ddot{\mathbf{y}} = \mathbf{C}_1\ddot{\mathbf{q}} - \mathbf{C}_2 \tag{2.5}$$

The control input vector $\mathbf{U}$ of dimension K can be added to the generalised input vector with a coefficient matrix $\mathbf{B}$ specifying the location of the control action. There are various ways of formulating an inverse dynamics model as discussed in [52], but the

general formulation can be written as follows by moving the unknown control input vector to the left hand side in Eq. (2.5) and using the desired motion as additional constraint equations:

$$\begin{bmatrix} \mathbf{A} & \mathbf{F}^T & \mathbf{B} \\ \mathbf{F} & \mathbf{0} & \mathbf{0} \\ \mathbf{C}_1 & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \lambda \\ \mathbf{U} \end{bmatrix} = \begin{bmatrix} \mathbf{D}_1 + \mathbf{Q} \\ \mathbf{D}_2 \\ \ddot{\mathbf{y}} + \mathbf{C}_2 \end{bmatrix} \tag{2.6}$$

The algebraic differential equations in Eq. (2.6) can be solved for the second derivatives of the generalised coordinates, which can then be double integrated to obtain the motion of the system. The required control input vector and the Lagrange multipliers are automatically calculated during this process.

The mathematical procedure detailed above forms the core of the mathematical simulation engine of DYSIM. A flow chart summarising the method of functionality of DYSIM is shown in Fig. 2.8.

To initiate the generation of a numerical model using DYSIM, the user only has to provide DYSIM with elementary physical and dimensional data regarding the makeup and construction of the mechanism to be modelled. The processes of deriving constraint equations, equations of motion and Jacobian expressions are carried out automatically by the program using this data. DYSIM also creates a list of variables which makeup the system. With the model generated, the user also has the capability to specify and define additional *user defined functions* (UDF's) to further enhance the model. These functions allow for the presence of external inputs to the system, such as friction or a reactive force from a load to be modelled. The initial conditions of the system must now be defined. These initial conditions take into account not only initial link positions but also initial link velocities. The user must first define the dependent and independent variables, before providing DYSIM with some initial estimates for mechanism link positions as well as the initial rate of change (velocity) of the independent variables. Using this data, DYSIM is capable of carrying out calculations to accurately calculate

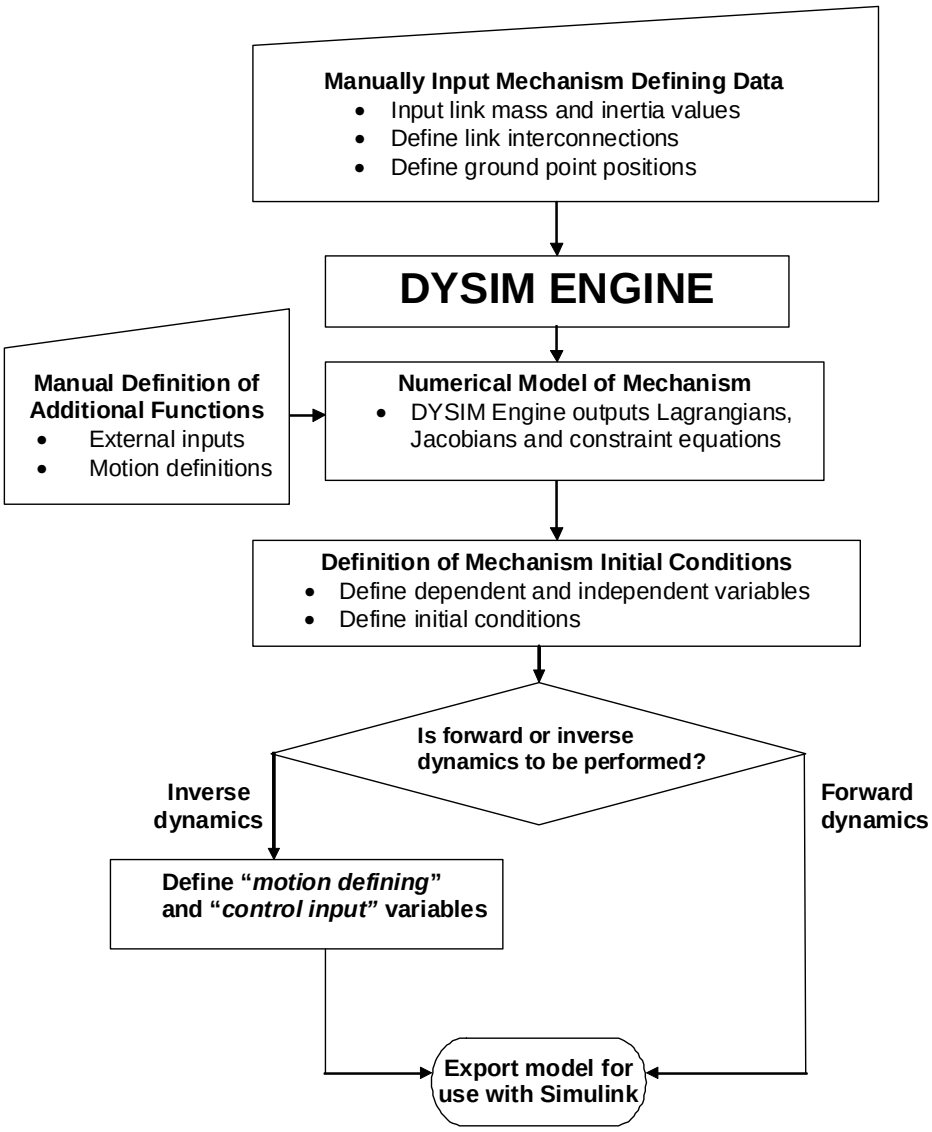


Figure 2.8: The method of operation of DYSIM

the positions and rates of change of the dependent variables, using the estimates as guides to steer the process.

Lastly, the user must decide whether the model to be generated should be in the form of a forwards or inverse dynamics simulation. If an inverse dynamic model is desired then the user must specify the *control input* and *motion defining* variables. With the model now fully defined, DYSIM exports the model to the MATLAB/Simulink environment.

2.2.2 The DYSIM User Interface

To interact with the user, DYSIM possesses an intuitive graphical user interface. This interface presents the user with the ability to input data and to select options to create the desired model. The process of creating a model of the Woodpecker mechanism using DYSIM will be discussed in this section.

Having loaded DYSIM and created a new project file, the user is presented with the *Planar (2D) Mechanism Interface* screen, depicted in Fig. 2.9. Using this screen the user has the ability to define the physical and geometric properties of the bodies making up the model, as well as specifying the pattern of interconnection between the bodies. The *Planar (2D) Mechanism Interface* screen depicted in Fig. 2.9 can be broken into a number of sections, each of which has its own specific functions.

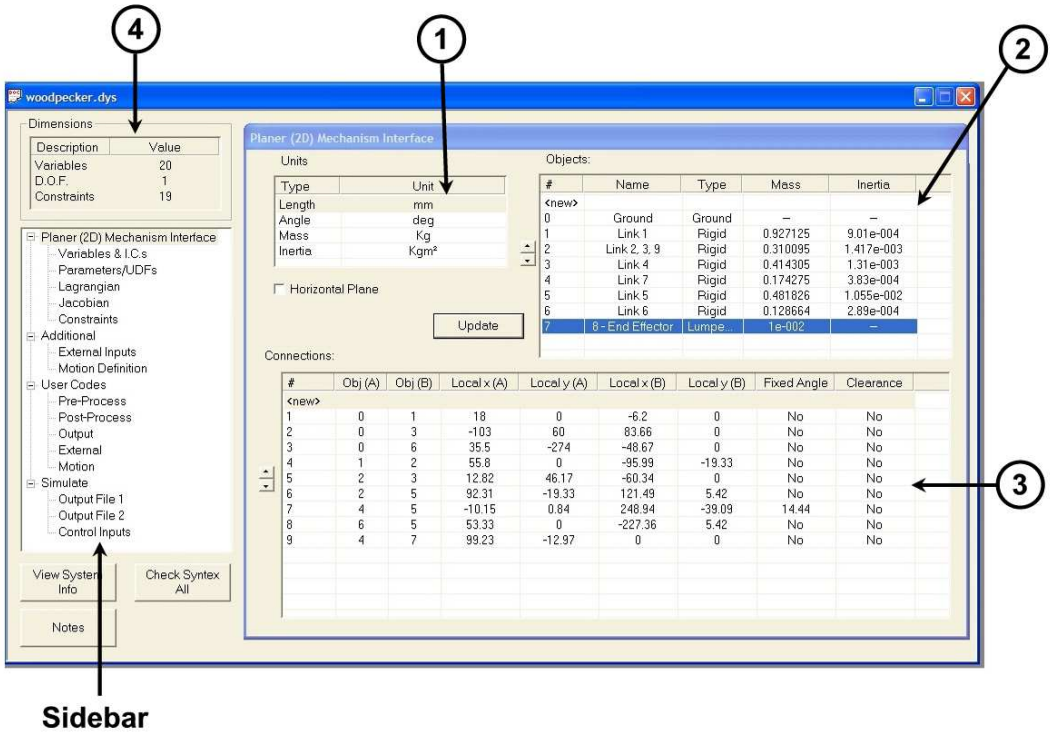


Figure 2.9: The Planar (2D) Mechanism Interface screen

Fundamental to the graphical user interface is the *sidebar* as labelled in Fig. 2.9. This section is always displayed and accessible by the user. The function of the sidebar is to allow navigation between the different parts of the program. To ac-

cess the *Planar (2D) Mechanism Interface* mechanism screen, the user must select Planar (2D) Mechanism Interface from the sidebar. The *Planar (2D) Mechanism Interface* screen consists of a number of different sections, each of which will be discussed in turn.

Section 1. Using the options in this section of the screen, the user can define the units of the parameter values to be entered. The user can also specify whether the system is to positioned in the horizontal or vertical plane and therefore whether gravitational effects should be considered in the resultant equations of motion.

Section 2. Using this section of the screen, the user defines the individual bodies which make up the system being modelled, which in this case are mechanism links. Typical links, referred to as *rigid bars* require the definition of mass and inertia data. Bodies can also be defined as point masses and inertias if required. Point mass bodies possess mass but no inertia. Similarly point inertia bodies possess inertia but not mass. DYSIM assigns each body a number, in this case 0 through to 7, which the user must use when referring to a particular body in the latter stages of model creation. For ease of reference, the user can also assign a name to each body. In this case names have been assigned to reflect the labelling convention of the links as defined in Fig. 2.2. In all cases, DYSIM refers to the ground as *body 0*.

Mass and inertia values for each link were supplied to DYSIM as defined in Tab. 2.1. Note that the triangular tertiary link (links 2, 3 and 9) was defined as being a single body, *body 2*. With the exception of link 8, all the links were defined as being rigid bars. Link 8 is defined as being the theoretical end effector of the mechanism and is therefore a theoretical point in space. It was defined in DYSIM as being a *point mass* with a negligible mass.

Section 3. Using this section of screen, the user defines the manner in which the individual bodies are connected together. To specify a connection, the user must define the distance between centre of mass of the first body, body A, to the point where it connects to the second body, body B. Correspondingly, the user must also specify

the distance from the centre of mass of the second body to the connection point with the first body. Distances are specified using in terms *body fixed* coordinate system local to each body. The origin of the body fixed coordinate system for a given object is the centre of mass of that given body. In this way, all connection points are defined as coordinate pairs as shown in Fig. fig:prototype-Dysim1. DYSIM assigns each connection an identification number, in this case, 1 through to 9. A pictorial example of the definition of *connection 4*, the connection between link 1 (body 1) to the triangular tertiary links 2, 3 and 9 (body 2), is shown in Fig. 2.10.

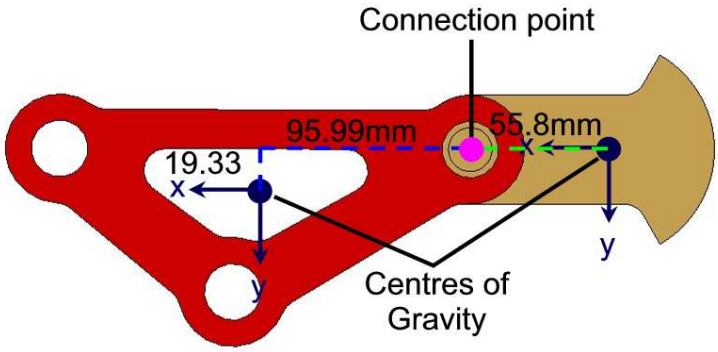


Figure 2.10: A pictorial definition of the connection between the link 1 and links 2, 3 and 9 showing local body fixed coordinate systems and distances to the connection point

Note that connections 1 to 3 are connections to the ground (body 0) with body A being defined as the ground. In these cases the coordinates specified for body A, represent the coordinates of the respective ground point in relation to the global coordinate system. In a similar fashion as before, the coordinates for body B represent the distance from the centre of mass of the grounded link to the ground point in terms of the body fixed coordinate system of the link in question.

On occasions, it may be desirable for the two links to be connected at a fixed angle to one another. An example of this can be seen in connection 7, the connection between links 7 and 5 (bodies 4 and 5). In this case, the links are assembled at an angle of 14.44° to one another. DYSIM provides a facility for this constraint to be implemented. Angles are taken with respect to the body fixed coordinate systems of both bodies and so are not dependent on initial conditions.

Section 4. This section of the screen possesses no user definable parameters but displays status messages regarding the mechanism under construction. In this section, DYSIM informs the user of how many variables have been automatically defined. The number of variables defined is dependent on the number of bodies defined. Rigid bar type bodies possess three variables; the absolute position of the centre of gravity of the body in x and y coordinates and angular orientation. The horizontal and vertical positions are defined as linear displacements of the origin of the body fixed coordinate system of a particular body with respect to the origin of the global coordinate system. Similarly, the angular orientation parameter is defined as the angular displacement between the body fixed coordinate system of the body and the global coordinate system. Point masses and inertias contribute only linear parameters i.e. vertical and horizontal positions. DYSIM adds an additional variable to represent *simulation time*.

This section of the screen also displays the number of positional constraints imposed on the system. These constraints are automatically derived through the definition of connections. The difference between the number of variables and number of constraints is the number of degrees of freedom (D.O.F) of the modelled system.

Using the data input to the system via the *2D Planar Mechanism Interface* screen, DYSIM generates a series of parameter values describing the masses and positions of the vertices of the defined bodies. The user can view these parameter values as well as defining additional *user defined parameters* (UDF's) parameters by selecting the *Parameters and User Defined Functions* in the sidebar. A screenshot of this screen is shown in Fig. 2.11

DYSIM also automatically generates the Lagrangian, Jacobian and constraint equations of the system. Although not user modifiable, the user can view the form of these functions by selecting the relevant item via the side bar. For illustrative purposes, the constraint equations derived for the model of the Woodpecker mechanism are shown in Fig. 2.12. The Lagrangian and Jacobian functions are displayed in the same fashion.

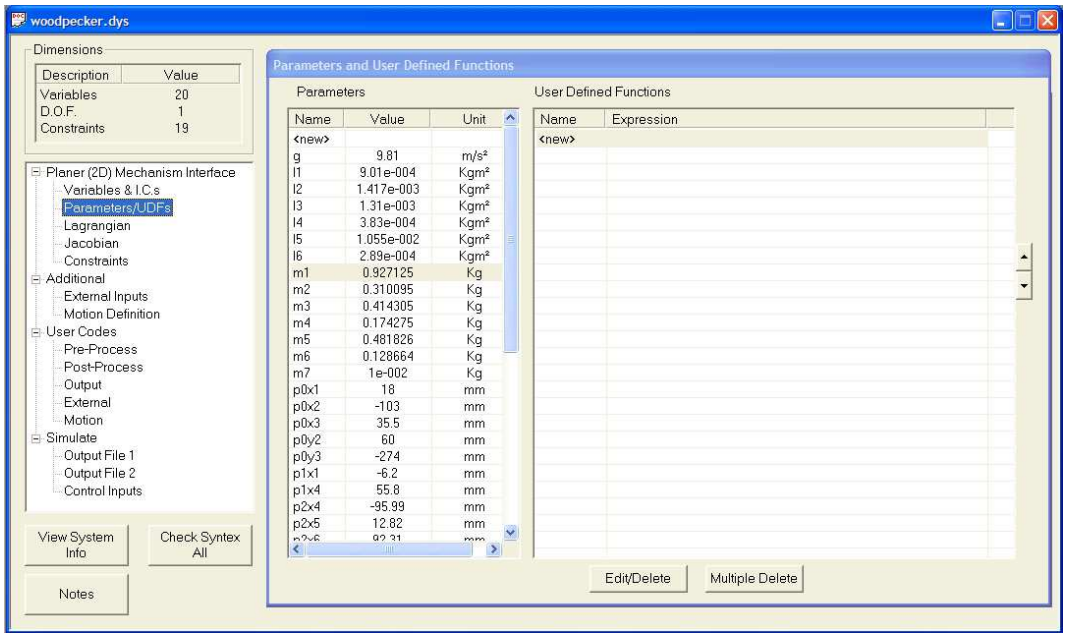


Figure 2.11: The Parameters and User Defined Functions screen

It can be seen that the constraint equations are expressed in terms of parametric constants, as shown in Fig. 2.11 and also in terms of variables, which are named numerically. In this case, they are named *v1* through to *v20*. The definition of these generalised variables is shown in the *Variables & I.Cs* (initial conditions) screen. A screen shot of this screen is shown in Fig. 2.13.

As well as giving each variable a label, DYSIM also gives each variable a name suffixed x for vertical displacements, y for horizontal displacements and a for angular displacements. The proceeding number shows to which body the variable relates. The specified unit for each variable is also shown. Initial variable values for each of these variables are displayed. First derivative values (i.e. velocities) for each of the variables are also displayed. Upon first creating a model, all variable values are set by default to 0 awaiting initiation by the user. To initiate the variable values, the user must first select which of the variables are independent variables. The number of independent variables must equal the number of degrees of freedom of the system. In this case, variable *v3*, the angular position of the link 1 (body 1) was selected as being the independent variable. The user must then input variable value for the initial values and for the first derivatives. If the user then clicks the *Auto Calculate* button, DYSIM will calculate

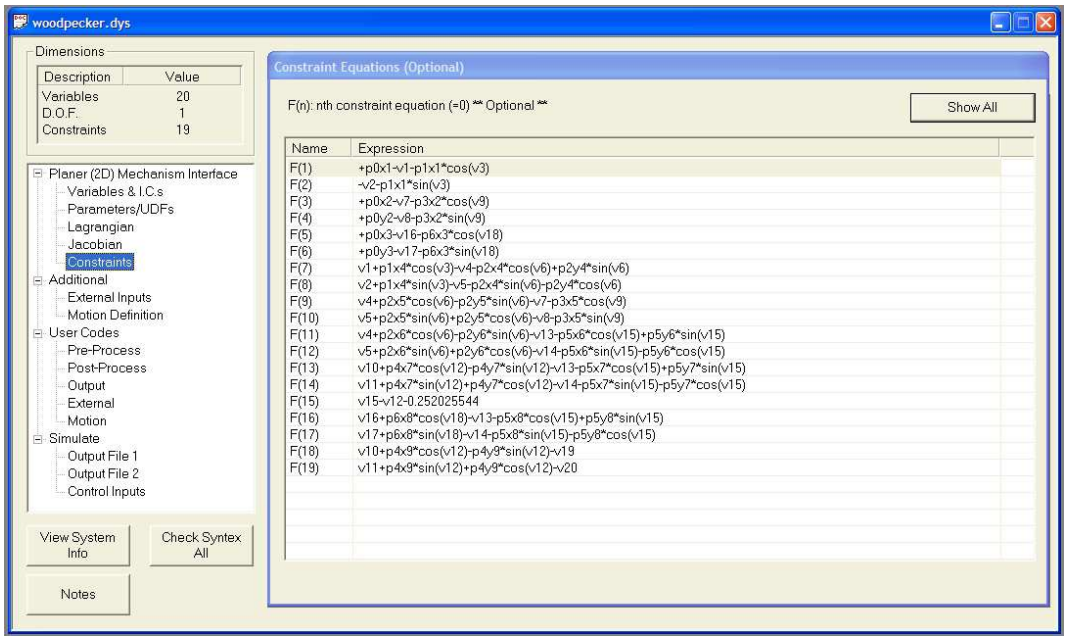


Figure 2.12: The Constraint Equation screen

initial values for the other, dependent variables and derivatives using the constraint equations. Clicking the *View* button will display a schematic view of the model in the orientation dictated by the initial conditions. With some mechanism designs, it may be possible for the mechanism to legitimately assume multiple orientations for a given value of independent variable. To ensure that the mechanism is assembled correctly, estimates of the values of the dependent variables in the desired orientation can be provided to steer the assembly algorithm towards the correct configuration.

If the user wishes to implement additional functions to the system, the option to do so is provided via the *Additional Functions* option in the sidebar, accessible by clicking *Additional*.

With the model now fully defined, the user must lastly decide whether a forwards or inverse dynamic simulation is desired. This can be done by selecting the *Simulation* option on the sidebar. A screenshot of this screen is shown in Fig. 2.14.

Using this screen, the user can select the type of desired simulation. If inverse dynamics is required, the user must also specify the control and motion defining variables. In the

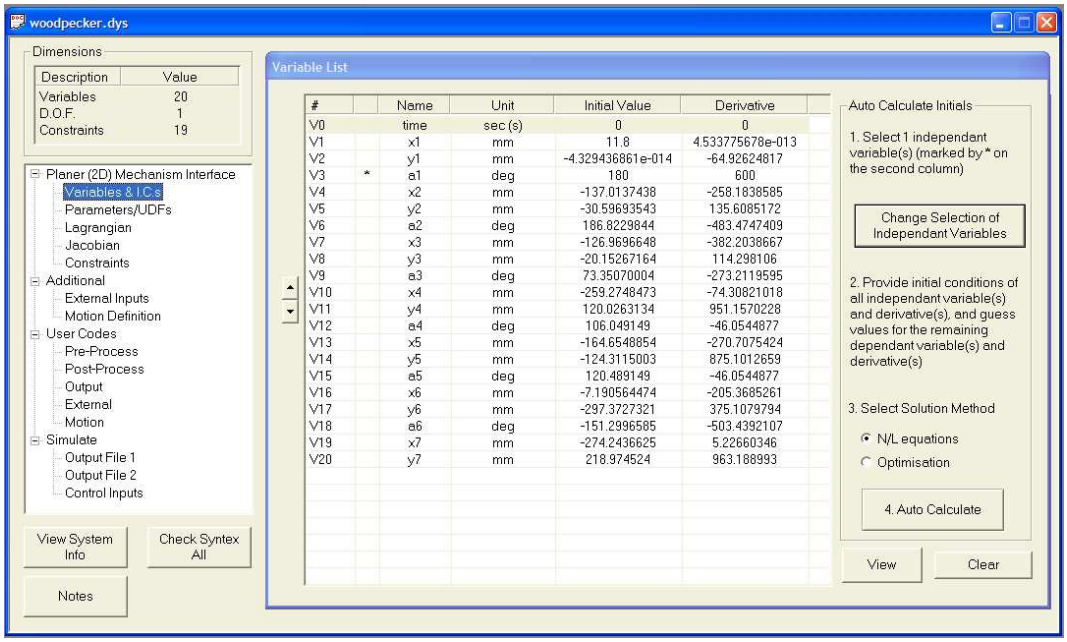


Figure 2.13: The Variables and Initial Conditions screen

case of the Woodpecker mechanism model, it was desired that variable 3, the angular position of link 1 should be both the control input variable and motion defining variable as shown in Fig. 2.14. Lastly, the user must click the *Create* button to export the model for use within the MATLAB/Simulink environment.

2.2.3 Preliminary Analysis of the Mechanism and Model Verification

Using the DYSIM model of the Woodpecker mechanism, the mechanism was analysed using inverse dynamics. Figure 2.15 shows the model used to perform the inverse dynamic simulation.

The inputs to the model allow the user to specify additional external inputs and motion definitions to the model. Signals passed into the model using these have an identical effect to specifying external inputs using *User Defined Functions* section of DYSIM, accessible by selecting, *Parameters/UDFs* on the sidebar. With simple constant velocity motion desired, no additional inputs to the model were necessary and so these inputs were set to zero. Using this model, it was possible to simulate the motion of

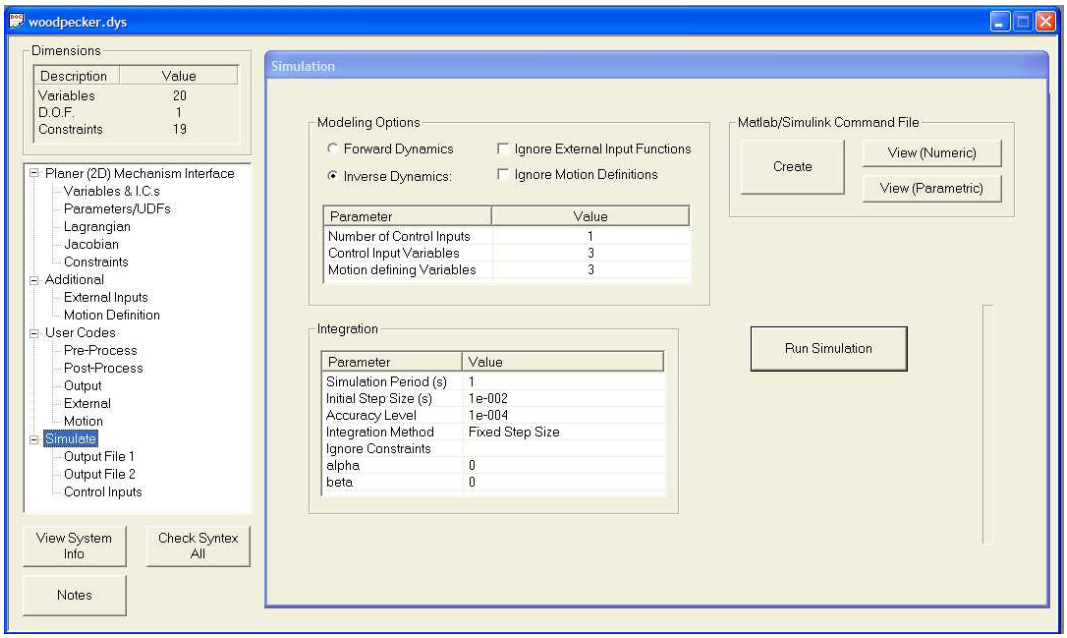


Figure 2.14: The Simulation screen

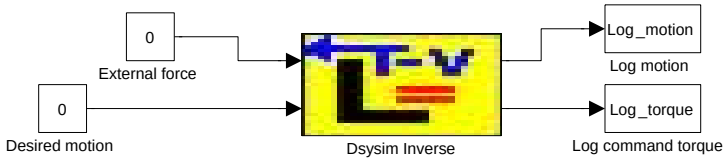


Figure 2.15: The Simulation screen

the mechanism when the crank is cycled at a constant velocity as well as predicting the variation in torque which would be needed to be exerted on the mechanism crank to generate this motion. Initially, particular interest was paid to the form of the orbit of the end-effector of the mechanism over a single cycle of crank. The resultant orbit is shown in Fig. 2.16.

The form of simulated orbit shown in Fig. 2.16 was compared to the form of the orbit shown in the original schematic of the mechanism as shown in Fig. 2.3. The two orbits were highly comparable, indicating that the geometry of the mechanism had been correctly modelled.

Also of particular interest was the simulated variation in torque which the drive motor would need to exert on the mechanism crank to propel it at a constant speed over an entire cycle of crank. The mechanism was simulated operating at cyclic speeds of 100,

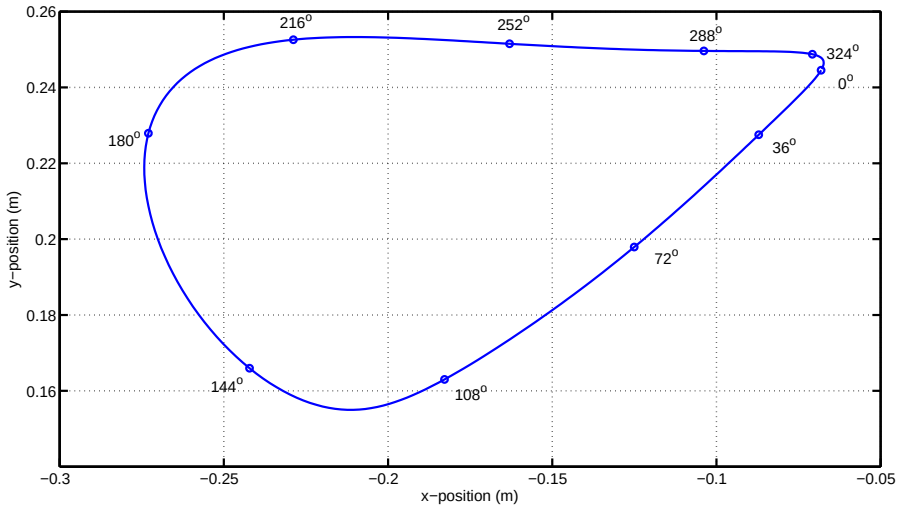


Figure 2.16: The simulated orbit of the end effector of the Woodpecker mechanism

200 and 300 rpm. The simulated variations in torque were plotted against crank angle, as shown in Fig. 2.17.

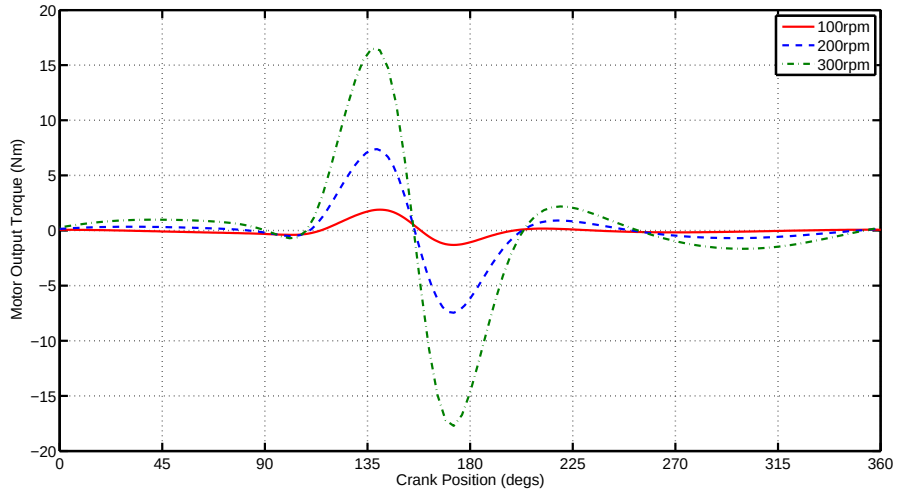


Figure 2.17: The calculated torque required to drive the Woodpecker mechanism crank at a variety of constant velocities over a complete cycle

From Fig. 2.17 it can be seen that see that the overall shape of the torque plots do not vary with cyclic speed, with peaks and troughs occurring at very similar positions of crank angle. The peak-to-peak magnitude of these curves do however, vary increasing with cyclic speed. These peaks and troughs are a result of physical non-linearities and imbalances in the design of the mechanism. Of particular note is the presence of a

pronounced torque peak followed by a torque trough towards the centre of the cycle. In this region of the motion, the behaviour of the mechanism is the most non-linear. As discussed in Sec. 1.2, the amount of harmonic content present in the output motion of a mechanism is proportional to the peak-to-peak magnitude of the torque variation needed to generate that motion [27, 31]. It is therefore highly likely that the majority of the harmonic content in the output motion is induced in this region of the motion. In this region, the motor has to work the hardest to maintain a constant since it has to rapidly accelerate and then brake the mechanism to ensure constant speed operation. The variation in torque in the remainder of cycle is much smoother with less variation, indicating a more linear physical behaviour. Based on theses observations, work will be done to develop methods of minimising the magnitude of the torque peak and trough.

2.2.4 Modelling the Deva004 Motion Controller

A summary of the method of operation of the controller was detailed in the Deva004 User's Manual [57]. With reference to this document, as well as with additional information gained through conversation with the manufacturer of the Deva004 a full picture of the operating procedure of the Deva004, was developed. The DEVA004 card can only operate in *position control* mode. However, when active, the controller acts to monitor and control both the velocity and position of the motor. It does this using two separate PI control loops, one for velocity and one for position. An estimation of the architecture of the control strategy of the Deva004 is depicted in Fig. 2.18.

Analysis of Fig. 2.18 shows that the output of the controller is governed by two separate PI controllers of standard form [29, 58]. One controller governs the velocity response of the system, whilst the other governs position control.

With reference to Fig. 2.18, it can be seen that the Deva004 operates in the following, iterative manner to control and regulate the motion of the motor. Based on a user defined motion profile, a value of target position, P_{dem} is passed to the controller. This value is then compared to the actual position of the motor P_{out} generating a position

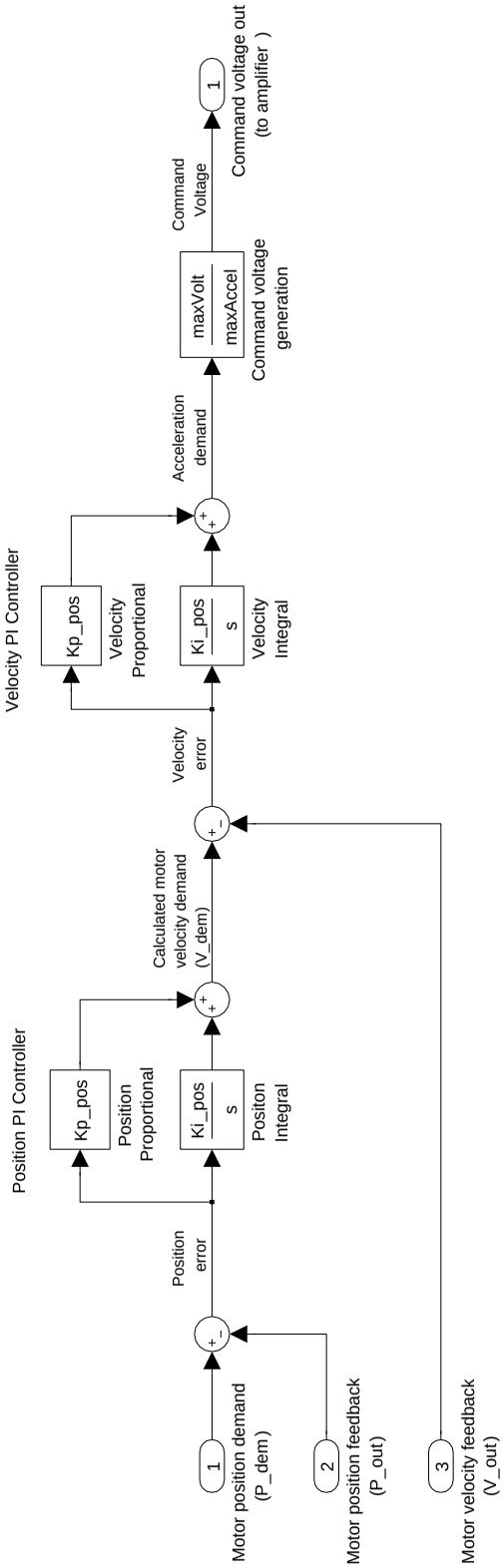


Figure 2.18: The control architecture of the Deva004 motion control card

error value, P_{error}. This value is then passed into the position PI controller. The position proportional gain value, Kp_{vel}, is defined as having units s^{-1}, where s is the unit for time in seconds. Thus with a position value as an input value, the output of the controller is expressed in the units $rads^{-1}$ i.e. angular velocity. The controller interprets these velocity values as being an *internal velocity demand* signal, V_{demInt}. This signal is compared to the actual velocity of the motor, V_{out} to create a velocity error value, V_{error}. This signal is passed through the velocity PI controller. Again the velocity proportional gain term has the units s^{-1} thus its output has the units $rads^{-2}$ i.e. angular acceleration. This output signal, A_{demInt}, is scaled according to two user definable parameters $maxVolt$ and $maxAccel$, to generate a final output voltage signal. This output signal is passed out of the Deva004 to the BRU-DDM30. The parameters $maxVolt$ and $maxAccel$ refer to the maximum permitted voltage which the Deva004 can output and the maximum permitted acceleration which the controller can demand respectively. The values of these parameters were selected through the process of configuring and setting up the Deva004 control card. This process is described in the Deva004 User's Manual [57].

2.2.5 Deva004 Operational Logic

In addition to modelling the dynamics of the Deva004 control card, it was also necessary to carry out work to understand operational logic with which the device operates. It was important to not only understand how the controller responded to simple demand signals, such as step and ramp signals but also to more complex variable velocity type signals. This information was not made readily available in the Deva004 documentation. Instead, a series of tests were performed using a software emulator of the Deva004 included with the Deva004 Software Developers Kit [55]. The emulator simulates the behaviour of a Deva004 controlling a generic, theoretical motor. Control of the emulator can be performed using identical control scripts, as would be used to control a real motor. Using this emulator, a variety of command signals were invoked to allow the response of the controller to both simple and complex demand signals to be analysed.

With reference to the Deva004 Programming Guide [59], a method of defining complex position dependent, variable velocity profiles was developed, with the intention of the controller achieving each target velocity at each corresponding reference position. Such a style of motion is also known as *velocity cam function*.

Through these tests, a good picture of the behaviour of the controller was obtained. As previously discussed, when a motion command is passed to the controller, the controller processes this command and generates its own internal velocity command signal, V_{dem}, which is converted to an acceleration demand signal, A_{dem}, which is then scaled and output to the motor amplifier as a voltage. Two user defined parameters, $maxAccel$ and $maxDecel$, dictate the maximum permissible motor acceleration and deceleration with which the controller can command the motor to move. For simplicity both variables were set to equal values. For this reason, reference is only made to $maxAccel$ in the model of the controller. The operational logic implemented by the system dictates the way in which the controller forms the motor velocity command signal, V_{dem}.

When responding to a motion command, the controller creates its own internal model of a virtual motor with perfect system dynamics. This virtual motor always accelerates and decelerates at the full value of $maxAccel$ and $maxDecel$. The controller continuously calculates the displacement and velocity of this virtual motor. It passes signals to the real motor to command it to attempt to follow this virtual motor as closely as possible.

For a given motion command, if the virtual motor achieves the target speed before it has reached its target position, the virtual motor will remain at the target speed until it has reached its target position. If the motion under consideration is a cam function, then the next stage of the cam function will be invoked and so on. If the virtual motor achieves its target position before it achieves its target velocity, then the next stage of the cam function will be invoked at the correct position regardless of motor velocity.

To conclude a motion, the controller commands the virtual axis to come to rest at the final target position. To achieve this, the controller appears to monitor the motion

command list and detects the final target position. Based on the motion command, it calculates when and where in the motion a motion command should be invoked to bring the virtual motor to a standstill at the final target position decelerating at $maxDecel$. When the correct position has been reached, the controller invokes the command overriding the cam function.

It is important to reiterate that the controller passes signals to the real motor to allow it to follow the motion of the virtual motor as closely as possible, governed by the dynamics of the velocity and position PI controllers. Thus in order for the real motor to be able to follow a cam function well, the PI controller must be properly tuned and values of $maxAccel$ and $maxDecel$ set to sufficiently high values.

2.2.6 Modelling the BRU-DDM30 Servo Drive and S4075 Servomotor

The BRU-DDM30 was configured to operate in torque control mode. In this mode, the device acts to respond to input command voltage signals from an external control device, in this case the Deva004, and to convert these signals to current demand signals which it passes to the S4075 servo motor. The factor of proportionality relating input command voltage and output command current is referred to as RR and can be defined by the user using BRU Master. The system gain of the controller is referred to as K_m. Thus a model of the BRU-DDM30 was created as shown in Fig. 2.19 .

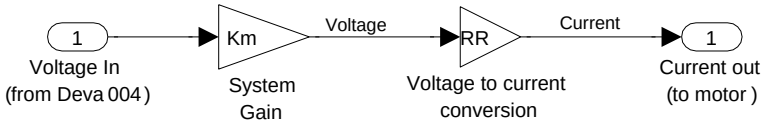


Figure 2.19: A model of the BRU-DDM30 servo drive

In response to this command current, the motor generates a motion with an output torque at its shaft directly proportional to the magnitude of the input current. The constant of proportionality relating command current to demand torque is referred to as K_{tq} and is a physical constant of the motor. Its value is stated in the technical specifications for the motor. A model of the S4075 servo motor was therefore created

as shown in Fig. 2.20.

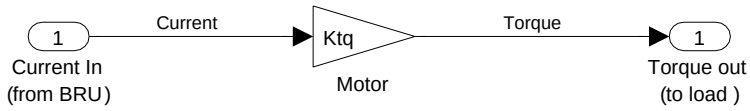


Figure 2.20: A model of the S-4075 servo motor

Since the Deva004 can only operate in position control mode, it was not possible to explicitly command the system to perform a *pure* velocity step or ramp type motion since a target position must also be specified. Instead, motions were defined which approximated to a velocity step or ramp motion within a sensible range of motion. In the case of a step motion, a motion was defined whereby the motor was commanded to accelerate from rest to the target speed with a very large acceleration. The target position was defined as being a long way away from the origin. Care was taken to ensure that the target position was far enough away to allow sufficient time for the system to achieve the target velocity and steady state conditions. To synthesise a ramp signal, a similar motion was demanded albeit with a more gradual acceleration. A target position far enough away from the origin was specified to permit sufficient time for behavioural trends to be observed.

This phenomenon was mimicked in the model of the controller, whereby all motion demand signals had to be expressed in terms of position demand signals. So for example, a velocity step demand signal had to be expressed in terms of a constant gradient position ramp signal and so on.

2.2.7 Combining the Models

The models of the Deva004, BRU-DDM30, S-4075 and the Woodpecker mechanism were all combined in a single model. Elements representing the belt drive transmission were included as simple gain terms of magnitude N. The complete model is shown in Fig. 2.21.

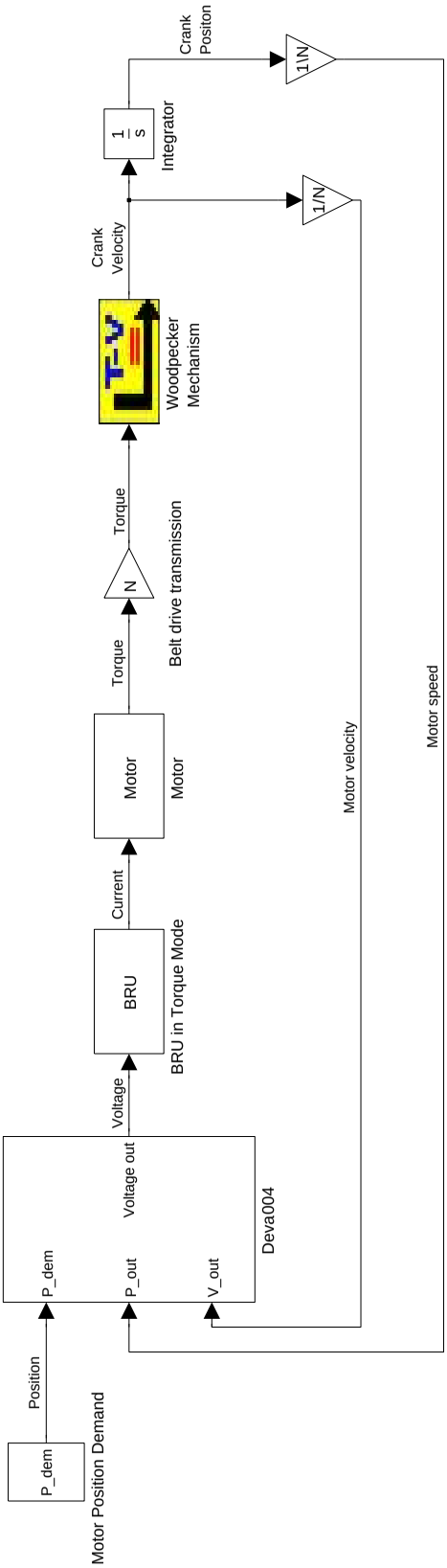


Figure 2.21: The control architecture of the complete Woodpecker mechanism test rig

The system was also modelled in a simplified configuration, in which the mechanism crank was disconnected from the rest of the mechanism. In this configuration *only* the mechanism crank is propelled by the motor. The crank is assumed to be a simple, linear load in this configuration. The model of the system in this configuration is shown in Fig. 2.22.

During actuation, the motor must do work not only to propel the mechanism but also the belt drive transmission and its own internal moving parts. Parameters were implemented in the model to account for this. In this model, the term J_{drive} represents the inertia constant and coefficient of friction of the crank, the motor and belt drive transmission acting on the system at the point of interface between the crank and the motor via the belt drive. The belt drive transmission consists of a drive belt and drive pulleys. The term b_{comb} represents the average coefficient of friction acting on the complete test rig, which in this case includes the crank, motor and transmission system only.

When considering the test rig, complete with Woodpecker mechanism, these parameters were incorporated into the model by modifying the DYSIM model of the mechanism. To simulate the effects of friction in the transmission, an extra expression was included in the model which considers the effective friction induced torque, $T_{friction}$, which acts on the system in opposition to driving torque from the motor. This torque was considered to act on the crank at the position where it interfaces with the motor. Taking into account the gear ratio, the magnitude of this *friction torque*, $T_{friction}$ is:

$$T_{friction} = b_{comb}\dot{\theta}_{crank}N^2 \tag{2.7}$$

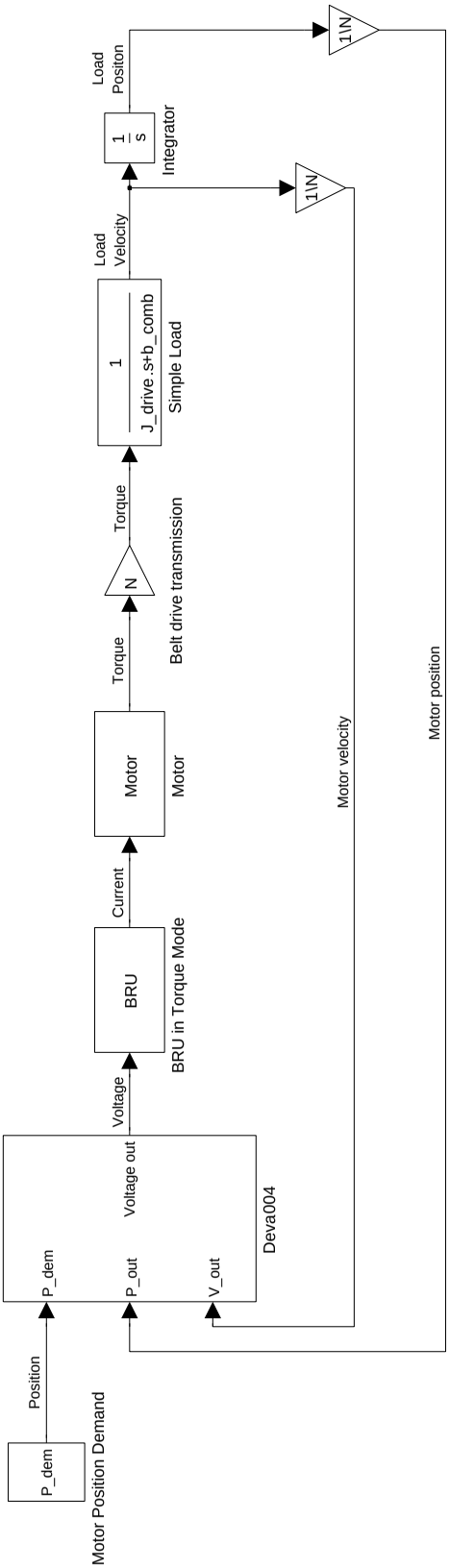


Figure 2.22: The Woodpecker mechanism test rig in its simplified configuration with mechanism disconnected from it crank

2.3 Practical Testing

With reference to the models derived in Sec. 2.2 it can be seen numerous system parameters exist which can affect the behaviour of the test rig during operation. The majority of these parameters are user definable, such as the definition of the velocity and position PI controller gains in the Deva004. In this way, the user can tune the behaviour of the system to suit an application. Other values are unknown physical constants, in particular J_{drive}, b_{comb} and K_m. In order to fully model the test rig, experimental work was carried out to obtain values for these parameters. Other system parameters exist which are user definable, allowing the user to tune the behaviour of the system, such as the definition of the velocity and position PI controller gains in the Deva004.

To estimate values of J_{drive}, b_{comb} and K_m the test rig was excited using a series of velocity step and velocity ramp style moves with various combinations of controller proportional and integral gain values. Derivative control was not used. The time responses of the rig to these motions were then analysed. The test rig was tested both its complete and simplified configurations.

With reference to the Deva004 Users Manual [57], suitable position proportional, Kp_{pos} and velocity proportional and integral gain values, Kp_{vel} and Ki_{vel} respectively were selected. Values were selected to achieve a system response which was quick, accurate and stable. In the two different system configurations, the dynamics of the system are clearly very different. Different gain setting combinations were therefore derived. The remaining PI gain values were set to zero.

2.3.1 Data Logging

In order to enable the behaviour of the test rig to be analysed it was necessary to measure and log the responses of the test rig during actuation. The following variables were logged against time:

1. Demand position P_{dem}
2. Output position P_{out}
3. Velocity command V_{dem}
4. Output velocity V_{out}
5. Motor drive command current I_{out}

The motor position and velocity demand and output variables P_{dem}, P_{out}, V_{dem} and V_{out} were all logged using the Deva004 card with a resolution of approximately 64 Hz. It was not possible to log data at a higher frequency than this due to performance limitations of the control card. These variables were derived by the Deva004 using encoder data, fed directly back to the device from the motor via the BRU-DDM30.

The variation of motor drive current I_{out} was logged using the DAQ via one of the two analogue outputs configured using BRU Master. The DAQ was able to sample this data at 1 KHz. The magnitude of torque output by the motor at its drive shaft is proportional to the current command signal with constant of proportionality being K_{tq}. A value for K_{tq} is specified on the motor data sheet. With this information it was possible to calculate the motor output torque using the following formula:

$$T_{out} = I_{out} K_{tq} \tag{2.8}$$

For clarity throughout this document, the following convention is used in order to distinguish between simulated and experimentally logged responses. Simulated response will be given the additional suffix *Sim*, whilst experimentally logged responses will be given the additional suffix *Log*. So for example, the logged output motor position would be referred to as P_{outLog}, whilst the simulated output motor velocity would be referred to as V_{outSim}.

2.4 Parameter Estimation

Using the logged data, work was carried out to estimate values for J_{drive}, b_{comb} and K_m. The system was first tested in its simplified form, with only the mechanism crank being propelled. The crank was actuated using a 300 rpm (crank speed) step demand signal. The controller was configured such that only position proportional and velocity proportional control was active. The velocity and torque time response is shown in Fig. 2.23. Calculations were carried out using this response to gain initial estimates for the unknown parameter values. Using these values in the model of the test rig the simulated response of the system to the 300 rpm crank velocity step demand signal was derived. The simulated response was compared with the experimentally derived response. J_{drive} and b_{comb} were adjusted in an iterative to achieve a better degree of fit between the two responses.

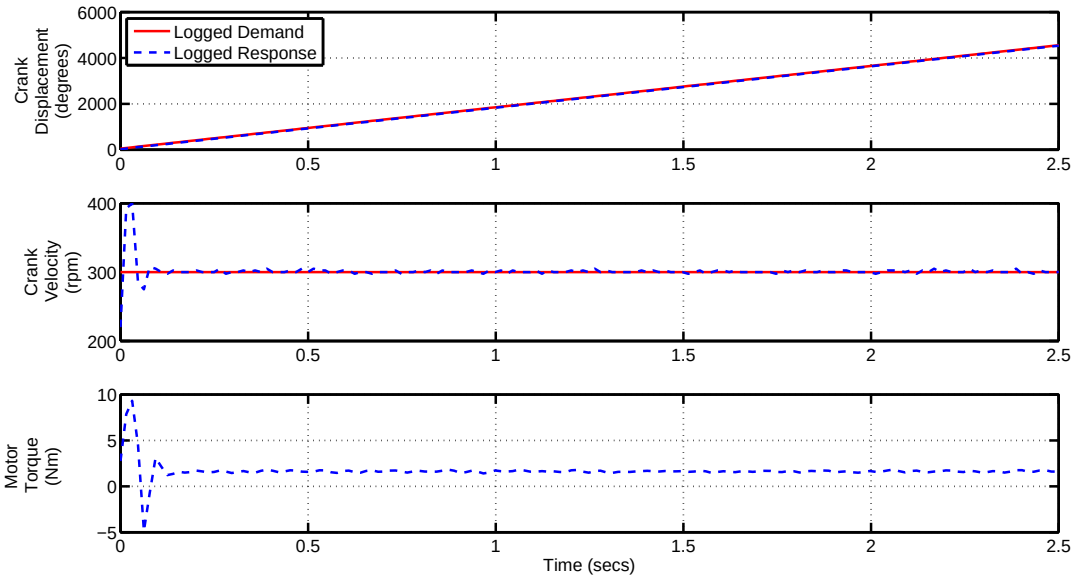


Figure 2.23: The velocity and torque responses of the Woodpecker mechanism test rig in its simplified configuration in response to a 300 rpm (crank speed) velocity step demand signal ($Kp_{pos} = 100$, $Kp_{vel} = 200$ and $Ki_{vel} = 0$)

2.4.1 Estimating The Controller Gain, K_m

To estimate a value for the controller gain value, K_m, an open loop model was derived containing the models of the Deva004, BRU-DDM30 and S-4075 only as shown in Fig. 2.24.

K_m was initially defined as being a unity gain. Values of P_{demLog}, P_{outLog} and V_{outLog} were fed into the model. Note that when incorporated in a full system, P_{outLog} and V_{outLog} would be motor status signals that would be fed back to the Deva004. With these variables logged, velocity and position feedback signals were fed into the controller in place of feedback signals thus the simulation is operating in an *open loop* configuration. In this way, with all input signals to the model of the Deva004 identical to those required by the real Deva004 for control purposes, then if the model of the system is correct, the output of the model should match the experimentally derived response.

The resultant simulated output torque of the motor T_{outSim} was compared to experimentally derived variation in output torque, T_{outLog}. K_m is the ratio between the two steady state responses, whereby:

$$K_m = \frac{T_{outLog}}{T_{outSim}} \tag{2.9}$$

2.4.2 Estimating the System Combined Coefficient of Friction, b_{comb}

As described by Sahinkaya, Rayner et al. [60] an estimate for coefficient of friction of a system of this type with only proportional gain active, can be obtained by analysing the steady state torque and velocity responses of a system. Whereby:

$$b_{comb} = \frac{N^2 \dot{\theta}_{out}}{T_{out}} \tag{2.10}$$

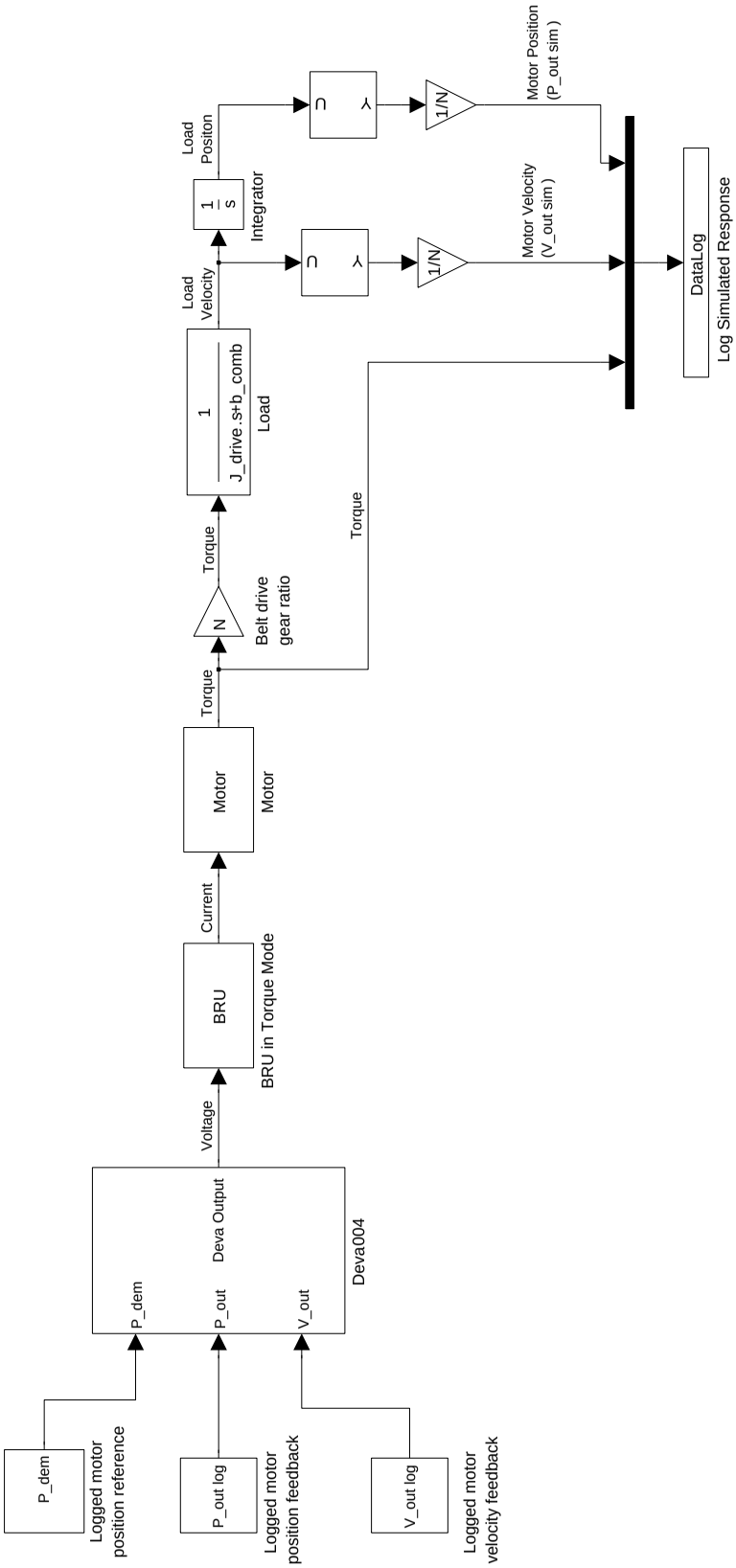


Figure 2.24: The open loop model of the Deva004 control card, BRU-DDM30 servo drive and S-4075 servomotor

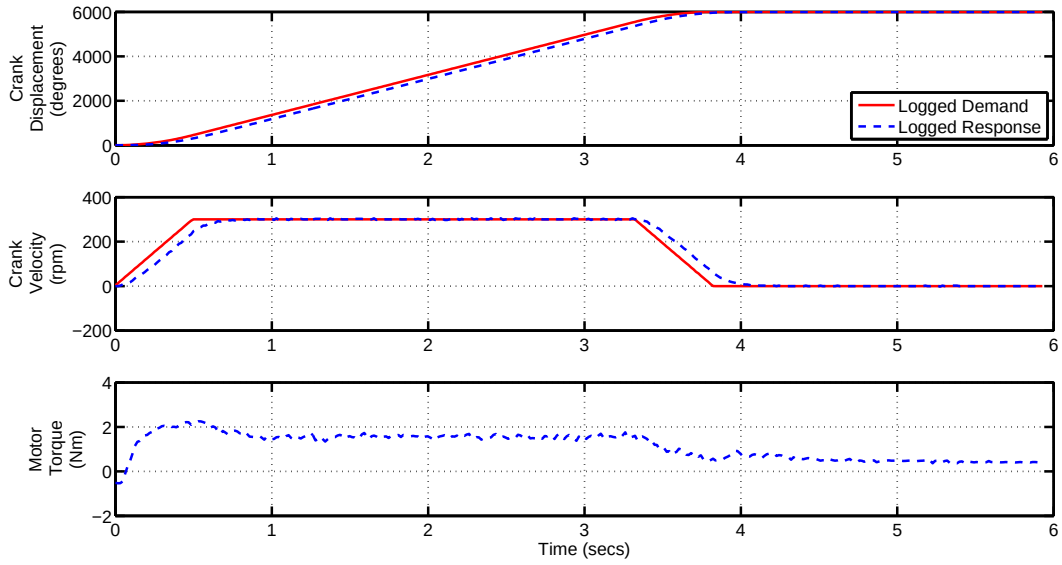


Figure 2.25: The velocity and torque responses of the Woodpecker mechanism test rig in its simplified configuration in response to a 300 rpm (crank speed) velocity ramp demand signal ($Kp_{pos} = 100$, $Kp_{vel} = 200$ and $Ki_{vel} = 0$)

2.4.3 Estimating the Transmission, Motor and Crank Inertia, J_{drive}

To estimate J_{drive}, the test rig was actuated in its simplified configuration, with only the mechanism crank being propelled, using a velocity ramp demands signal. Velocity proportional and position proportional gains only were enabled in the Deva004. The torque and velocity responses of the rig are shown in Fig. 2.25.

For a simple linear load, it is known that [60]:

$$T = \frac{b}{N}\omega + J\dot{\omega} \tag{2.11}$$

With reference to Eq. (2.11) it can be seen that if a load with inertia J and coefficient of friction b is accelerated at a constant linear rate using a velocity ramp demand signal with a constant propulsive torque T, then the velocity of the load, ω, will vary linearly with the propulsive torque, T. If a true ramp response is achieved the acceleration of the load, $\dot{\omega}$ will be constant.

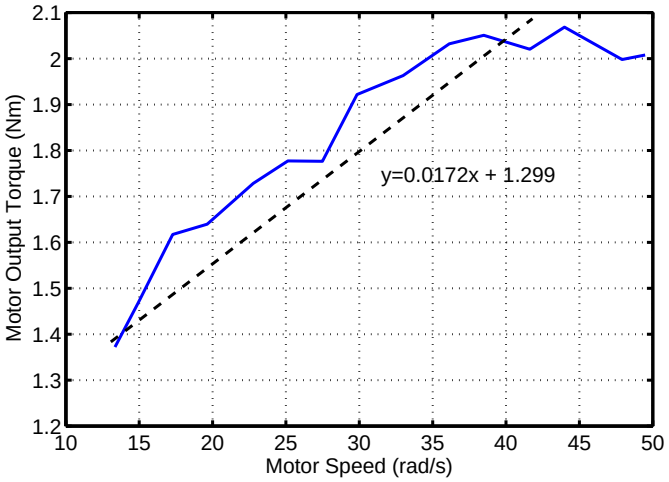


Figure 2.26: The velocity and torque responses of the Woodpecker mechanism test rig in its simplified configuration in response to a $54.8 rads^{-2}$ crank velocity ramp demand signal with a maximum crank velocity of 300 rpm ($Kp_{pos} = 100$, $Kp_{vel} = 200$ and $Ki_{vel} = 0$)

Thus if the variation in ω is plotted with respect to T, the resultant graph will take the form of a straight line. If the variables being used in this investigation are substituted for the generic variables in Eq. (2.11), the following expression is obtained [61]

$$T_{out} = \frac{b_{comb}}{N} V_{out} + J_{drive} \dot{V_{out}} \tag{2.12}$$

The test rig, in its simplified configuration, was actuated using a $54.8 rads^{-2}$ crank velocity ramp signal. Plotting motor output speed, V_{outLog}, against motor output torque, T_{outLog}, the response shown in Fig. 2.26 was obtained:

The straight line intersects the abscissa at $J_{drive}\dot{V}_{outLog}$ Nm. A value for J_{drive} can then be calculated directly. This method was applied to the logged data and a value of J_{drive} found.

2.4.4 Validating the Model of the Test Rig Its Simplified Configuration

Having now derived initial estimates for parameter values b_{comb}, J_{drive} and K_m, these values were inserted into the model of the test rig in its simplified configuration as shown in Fig. 2.22.

The logged position ramp demand signal, P_{demLog} corresponding to the 300 rpm crank velocity step signal, V_{demLog}, was fed into the model and the resultant simulated velocity and torque responses, V_{outSim} and T_{outSim} respectively, were analysed. The simulated responses were compared with the equivalent logged velocity and torque responses, V_{outLog} and T_{outLog} respectively. The values of b_{comb} and J_{drive} used in the model were fine tuned to achieve a better fit between the logged and simulated responses. Both the transient and steady state portions of the response were considered. Increasing b_{comb} was observed to have the effect of decreasing the magnitude of the steady state velocity response and increasing the magnitude of the steady state torque response. Decreasing b_{comb} had inverse effects. Increasing J_{drive} had the effect of slowing the rate of acceleration of the mechanism during the transient portion, increasing the settling the time of the system. Decreasing J_{drive} had the inverse effect. The final tuned response is shown in Fig. 2.27.

Thus:

$b_{comb} = 0.264$

$J_{drive} = 0.018\ kgm^2$

$K_m = 3.502$

2.5 Modelling of the Complete Test Rig

With values for all unknown system parameters now derived, the model of the test rig with the complete mechanism being propelled, depicted in Fig. 2.21, could now be

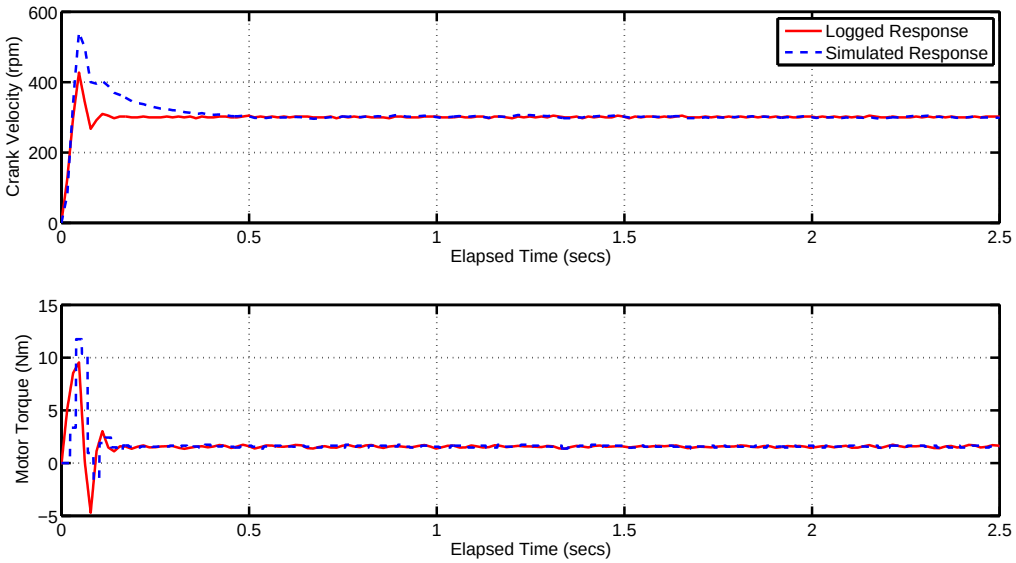


Figure 2.27: A comparison of the logged and simulated velocity and torque responses of the Woodpecker mechanism test rig in its simplified configuration in response to a 300 rpm (crank speed) velocity step signal ($Kp_{pos} = 100$, $Kp_{vel} = 200$ and $Ki_{vel} = 0$)

considered. The DYSIM model of the Woodpecker mechanism was modified as shown in Sec. 2.2.7 and the derived parameter values for b_{comb} and J_{drive} implemented. Before the model of the test rig could be used for further work, it was necessary to do work to validate its accuracy. To achieve this, the test rig in its complete configuration was excited using a 200 rpm (crank speed) velocity step demand signal. The response of the system to the demand signal was logged. Velocity integral (Ki_{vel}) and velocity proportional (Kp_{vel}) and position proportional (Kp_{pos}) active were enabled in the system.

The model depicted in Fig. 2.21 was then run using the pre-determined system parameters. The simulated and experimentally derived torque and velocity responses were compared. A poor level of fit between the two responses was achieved. To remedy this, the experimental work described in Sec. 2.4 was repeated to re-estimate values for b_{comb} and K_m. This time, the tests were carried out with the test rig in its complete configuration and using a 200 rpm crank velocity step signal.

With the mechanism in its simplified configuration, b_{comb} represents friction applied

to the system by the motor, transmission and bearings at the crank only. With the introduction of the mechanism, new sources of friction are introduced to the system, most notably in the bearings of the mechanism hence necessitating the re-estimation of this parameter. It may also be possible that the controller may respond to the change in load by varying its system gain thus it was deemed necessary to re-estimate this parameter also. It was not necessary to re-estimate J_{drive}, since this value represents a physical constant of the system.

2.5.1 Verifying K_m and b_{comb}

The model of the complete test rig depicted in Fig. 2.21 was modified such that the velocity and position feedback loops were removed. Logged position and velocity feedback data, P_{outlog} and V_{outlog} respectively were fed into the system in place of the feedback signals. This model can be seen in Fig. 2.28.

The model was first run with velocity integral control disabled. The pre-determined system parameters were used in the system. The steady state portion of the response was simulated and compared to the logged response of the system described previously. The magnitude of the simulated torque and velocity responses correlated poorly with the magnitudes of the response being too small.

K_m was recalculated using Eq. (2.9) and the method described in Sec. 2.4. The resultant gain value was larger. Running the simulation again using this new value for K_m gave a much improved correlation between the torque responses. The magnitude of the simulated velocity response however, was now too large indicating that the value of the system coefficient of friction, b_{comb} was too small. b_{comb} was increased in an iterative manner until a good level of correlation between the velocity responses was achieved.

Thus:

$K_m = 7.7694$

$b_{comb} = 0.3$

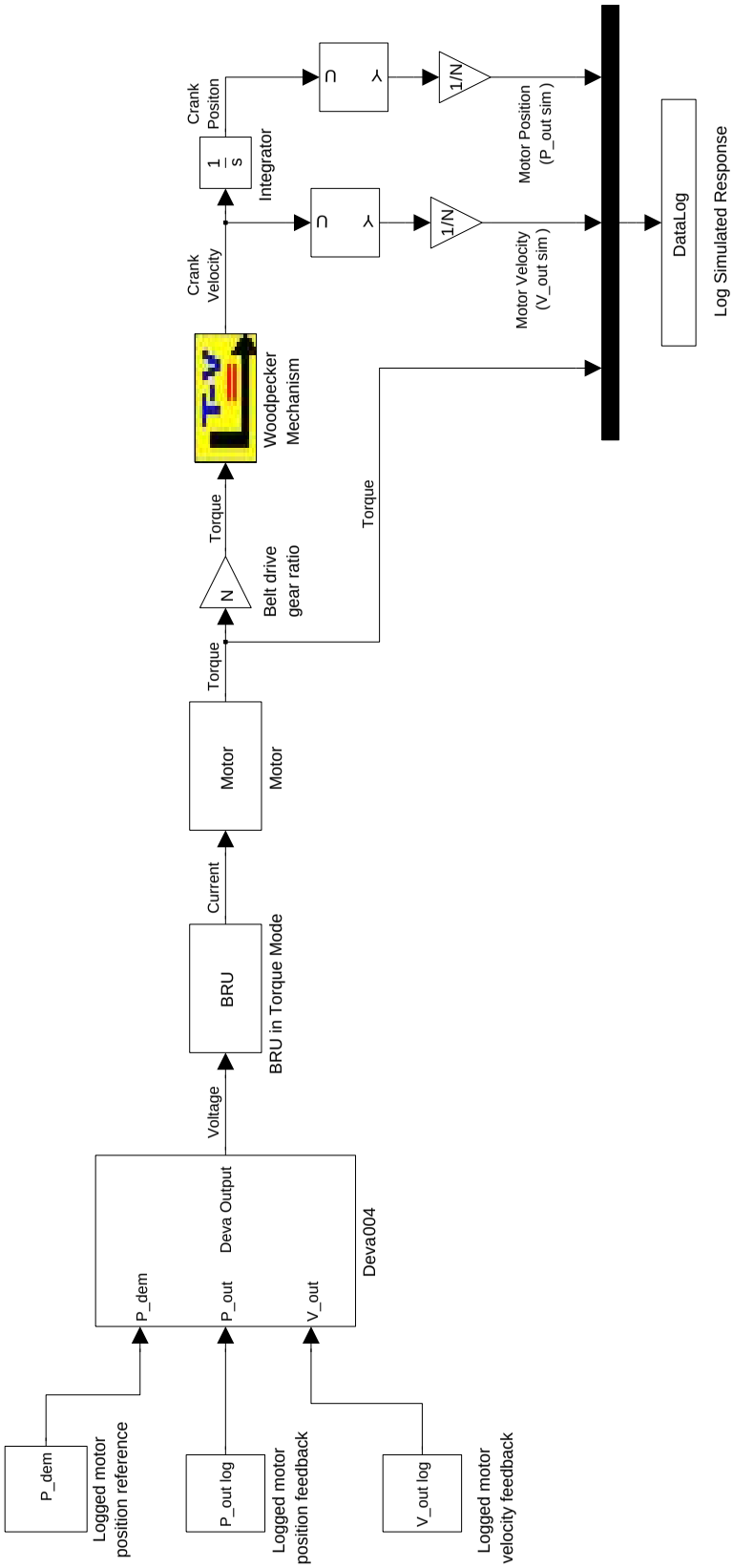


Figure 2.28: A model of the complete Woodpecker mechanism test rig with the feedback loop recreated using logged signals

Using these parameters, a simulated velocity and torque response was achieved that closely correlated to the logged response as shown in Fig. 2.29.

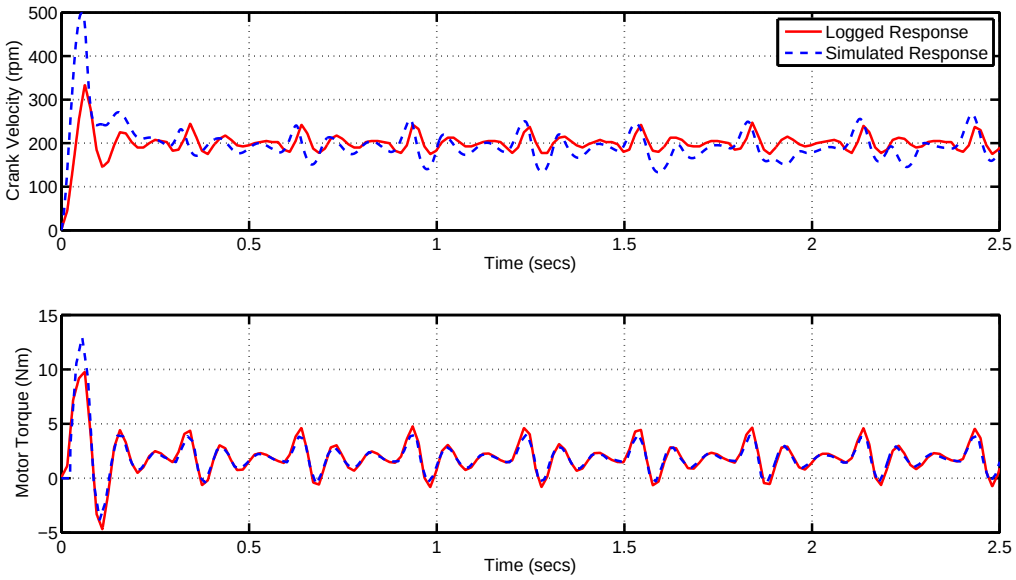


Figure 2.29: A comparison of the logged and simulated velocity and torque responses of the complete Woodpecker mechanism test rig in response to a 200 rpm (crank speed) velocity step demand signal ($Kp_{pos} = 100$, $Kp_{vel} = 200$ and $Ki_{vel} = 0$)

It was noted that the re-estimated value of b_{comb} was indeed larger than the previously derived value, confirming the hypothesis that the mechanism introduces new sources of friction to the system.

2.5.2 Verifying the Integral Control Loop

Velocity integral control was next enabled in the model, as shown in Fig. 2.28 and the simulation run again. Note that in this model, position and velocity feedback signals are supplied to the controller using logged data (i.e. the simulation is operating in an open loop configuration). A comparison between the logged and resultant simulated torque and velocity responses can be seen in Fig. 2.30.

From Fig. 2.30, it can be seen that the simulated torque and velocity responses ramp upwards with simulation time, deviating ever further from the logged responses. This

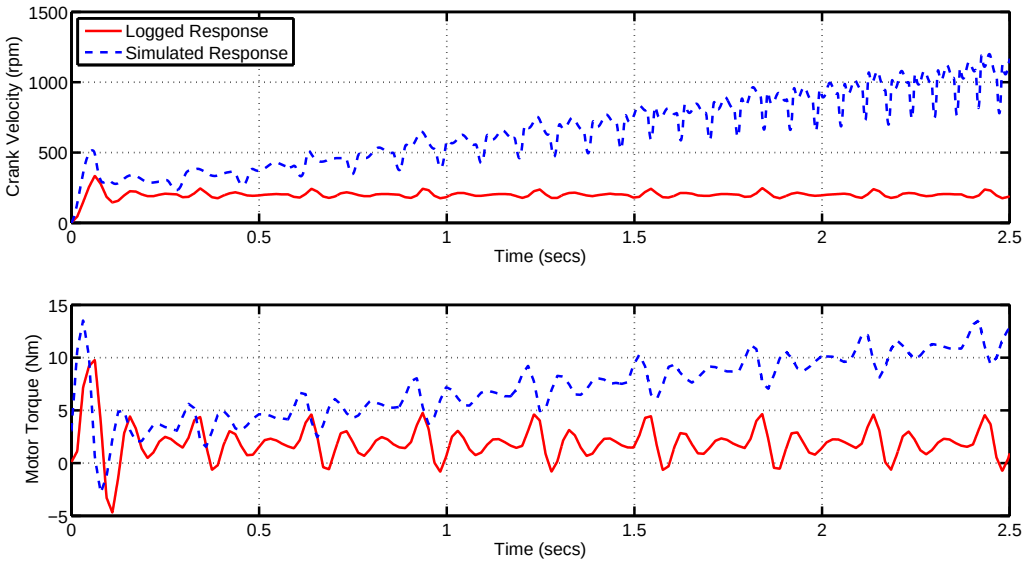


Figure 2.30: A comparison of the logged and simulated velocity and torque responses of the complete Woodpecker mechanism test rig in response to a 200 rpm (crank speed) velocity step demand signal ($Kp_{pos} = 100$, $Kp_{vel} = 200$ and $Ki_{vel} = 400$). (Note the presence of drift in the system responses caused by the integral controller.)

is due to the open loop nature of the simulation and the presence of errors and imperfections in the logged data.

Consider the manner in which the error term, V_{error}, at the input to the PI controller is calculated:

$$V_{error} = V_{dem} - V_{out} \tag{2.13}$$

With reference to Eq. (2.13), it can be seen that the two constituent signals used to calculate V_{error} are fed into the controller from experimentally logged data. Noise and other imperfections were observed to be present on these signals. These imperfections can lead to erroneous values of V_{error} being generated, resulting in unrealistic integral controller output signals and thus false system responses.

If the simulation is run in a closed loop configuration however, logged data is no longer used to form the feedback signals thus eliminating the aforementioned problems. The

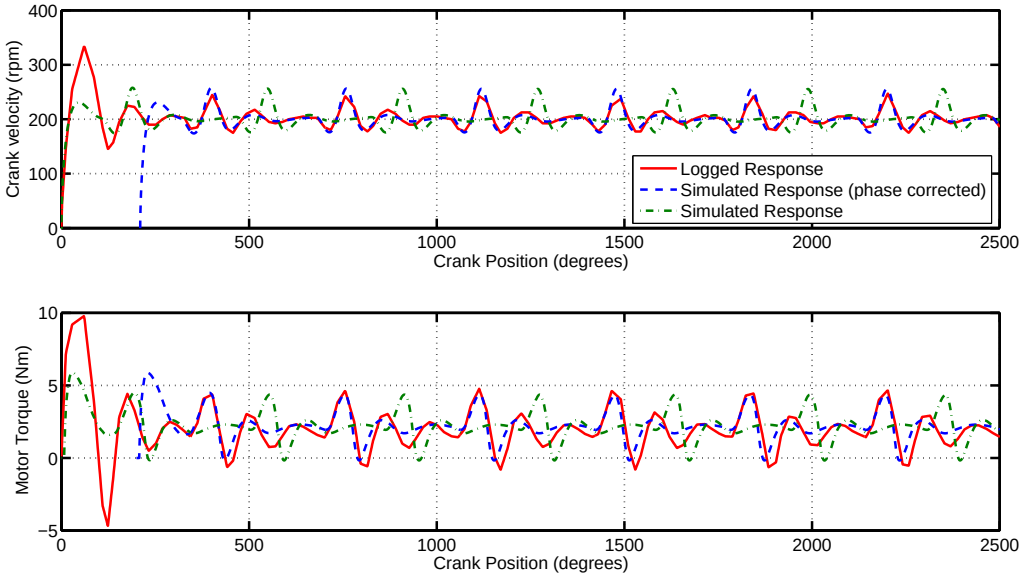


Figure 2.31: A comparison of the logged and simulated velocity and torque responses of the complete Woodpecker mechanism test rig in response to a 200 rpm (crank speed) velocity step demand signal ($Kp_{pos} = 100$, $Kp_{vel} = 200$ and $Ki_{vel} = 400$) ($Kp_{pos} = 100$, $Kp_{vel} = 200$ and $Ki_{vel} = 400$)

resultant simulated response, as shown in Fig. 2.31 was stable and free of the drift caused by the integral controller.

The system response shown in Fig. 2.31 has been plotted with respect to crank angle. It can be seen from this figure that the simulated steady state response lags the logged response by approximately 130 degrees of crank, despite initial configuration of both the simulated and real world systems being the same. The reason for this lag is unclear, but could be related to data logging issues. If however, this phase shift is compensated for by shifting the simulated response, the responses correlate very well. The simulated and logged characteristic steady state torque and velocity peaks and troughs correlate well in terms of magnitude, shape and frequency. Therefore it was considered that the model was good despite the phase shift.

2.6 Conclusions

In this chapter, work was carried out to create a model of a real world, prototype servomechanical system. At the heart of the system was a complex, multi-link mechanism with highly non-linear dynamic characteristics. It was propelled by an industrial servo motor via a belt drive and controlled by a motor drive unit acting as a slave signal amplifier, governed by an external motion controller. The system was observed to suffer from unacceptable levels of harmonic content when run at high speeds.

Using dimensional and physical data extracted from a CAD model of the mechanism, a numerical, planar model of the mechanism was created and verified. The mechanism was analysed using inverse dynamics to predict the variation in input driving torque that the motor must exert on the mechanism crank to propel the mechanism at a constant speed over the course of a complete cycle. Simulations were carried out to predict the behaviour of the mechanism during actuation. The mechanism was simulated being actuated at three different cyclic rates. A region in the motion was identified in which large harmonic content is likely to be induced. In this region, a sharp torque peak followed by a sharp torque trough is needed to propel the mechanism at constant speed, indicating a region of greatly non-linear behaviour. The amplitude of these peaks and troughs increased with running speed. The remainder of the torque response was far smoother by comparison. With reference to the literature, a torque response with a large peak-to-peak magnitude is said to be indicative of a motion, rich in harmonic content.

Using experimental data models of the belt drive transmission, drive unit and control card were also created. The models were combined to form a model of the complete test rig. Work was also done to validate the model, in which the simulated response of the test rig was compared to equivalent experimentally derived responses. Work was done to derive system parameter values and to refine the model. The velocity and torque responses of the final model to a 200rpm crank velocity step demand signal correlated well with the logged response to the same signal, particularly under steady-

state conditions. This implied that the model was valid and suitable for further work.

Chapter 3

The Cam Function Generation Method

In Ch. 2, a model of the a complete servomechanical test rig was developed. It was observed that the test rig had an steady state output motion that was rich in harmonic content when actuated at high speeds with a constant velocity demand signal. The cause of this harmonic content was identified as being due to significant non-linearities in the dynamics of the mechanism. When actuated using a constant velocity demand signal, the drive motor needed to exert highly variable amounts of torque on the mechanism crank. Over the course of complete cycle, this torque variation was observed to have a large peak-to-peak amplitude. The literature [27,31] indicates that such a torque profile is indicative of an output motion rich in harmonic content. It also states that reducing the peak-to-peak magnitude of a propulsive torque profile should reduce the amplitude of the harmonic content present in the resultant output motion. Work will be described in this chapter, in which a method of improving the dynamic performance of the system will be developed. The method will seek to reduce the peak-to-peak magnitude of the torque demands of the mechanism but without sacrificing system operating speed.

In this chapter, a method, entitled the *Cam Function Generation Method*, will be derived in which the dynamic performance of the test rig will be improved through the synthesis of a shaped velocity demand signal, referred to as a *cam function*, for use in place of the constant velocity demand signal. It is hypothesised that improved dynamic system performance can be achieved if the system is actuated using a velocity demand signal that is sympathetic to non-linear dynamics of the system. Such a signal would allow the non-linear characteristics of the mechanism to dominate the output motion in the regions of crank where they are most prominent, instead of the motion being rigidly controlled by the motion controller and servomotor. In other words, in the non-linear portion of the motion, the amount of influence which the drive motor can exert on the mechanism will be reduced, allowing the dynamics of the mechanism to dominate. The mechanism will therefore accelerate and decelerate according to its dynamic tendencies. Any discrepancies in cyclic rate can be compensated by the progression of the mechanism through the linear portion of its motion. By reducing the amount of influence the drive motor can have on the motion of the mechanism, the amount of torque which the motor can exert on the mechanism will be reduced. In this way, the peak-to-peak magnitude of the cyclic torque variation will be decreased, resulting in a reduction in the magnitude of the harmonic content in the output motion.

3.1 The Method

The development work for the *Cam Function Generation Method* was performed using the model of the complete test rig developed in Ch. 2. However, for clarity of explanation, the method will first be applied to a simple theoretical system in which a servo motor, controlled by a generic drive unit, propels the Woodpecker mechanism via a belt drive with a gear ratio N. A PI controller governs the system. The system operates in velocity control mode. Both integral and proportional control were enabled in the controller. The values of the controller gains were tuned to achieve a fast, stable response with minimal steady state error.

An element labelled, *Torque Truncation* was positioned between the motor controller and the belt drive. Using this element, it was possible to impose upper and lower limits on the amount of torque the motor could exert on the mechanism via the belt drive. A block diagram for this system is shown in Fig. 3.1.

Through the strategic imposition of torque truncation limits, it was possible to limit the amount of influence the motor could exert on the motion of the mechanism in regions where the torque limits are triggered. In such regions, the dynamics of the mechanism dominate the response motion of the system. Having made this modification to the model, a method was derived to synthesise a shaped velocity demand signal.

1. With torque truncation limits disabled, the response of the system to a constant velocity demand signal was first simulated to achieve a system response that satisfies the desired cyclic rate of the system. The resultant cyclic torque response of the system was analysed.

2. Torque truncation limits were selected, which truncate the peaks and troughs of the steady state torque response of the system to the constant velocity demand signal, isolating a narrow torque band.

3. With these torque truncation limits in place, the constant velocity signal was fed into the system once again. The resultant velocity response was logged against crank position. This signal forms the **velocity cam function**.

4. With torque truncation now removed, the velocity cam function was fed into the system as a crank position dependent velocity demand signal. This is shown diagrammatically in the block diagram in Fig. 3.2.

5. The steady state velocity response of the system to the velocity cam function was analysed and the average velocity of the mechanism crank over a complete cycle calculated. If the resultant average cyclic velocity value is unacceptably different from the desired cyclic velocity, the method is repeated from Step 2 onwards using different value torque truncation values, until a cam function is generated

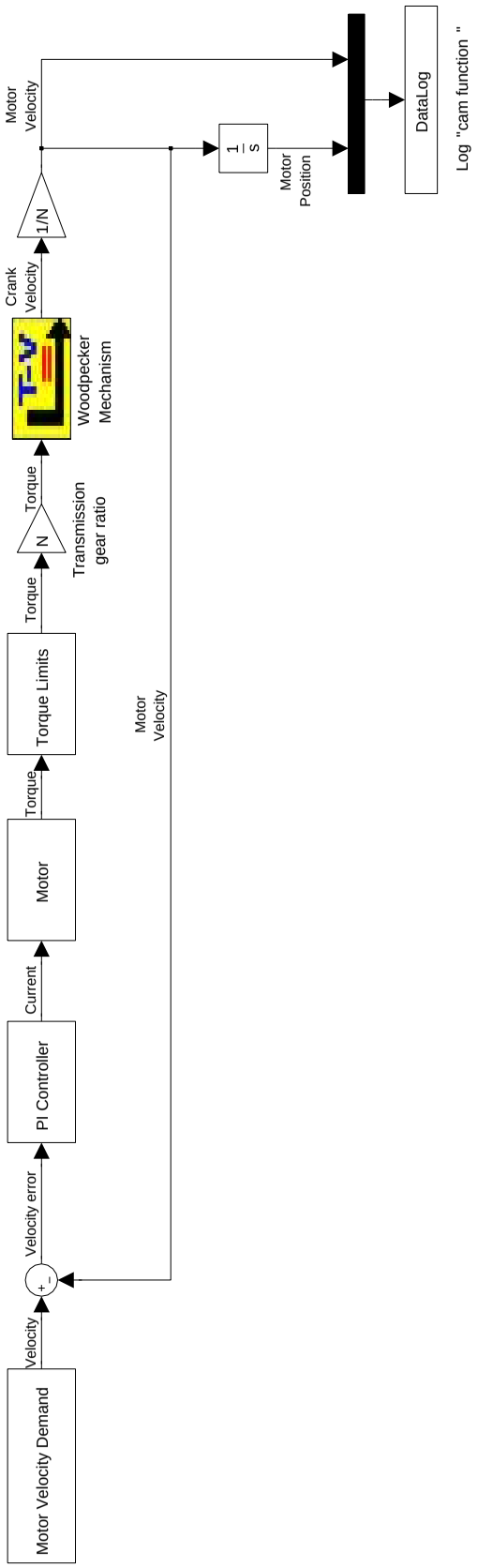


Figure 3.1: The theoretical model of a test rig incorporating the Woodpecker mechanism

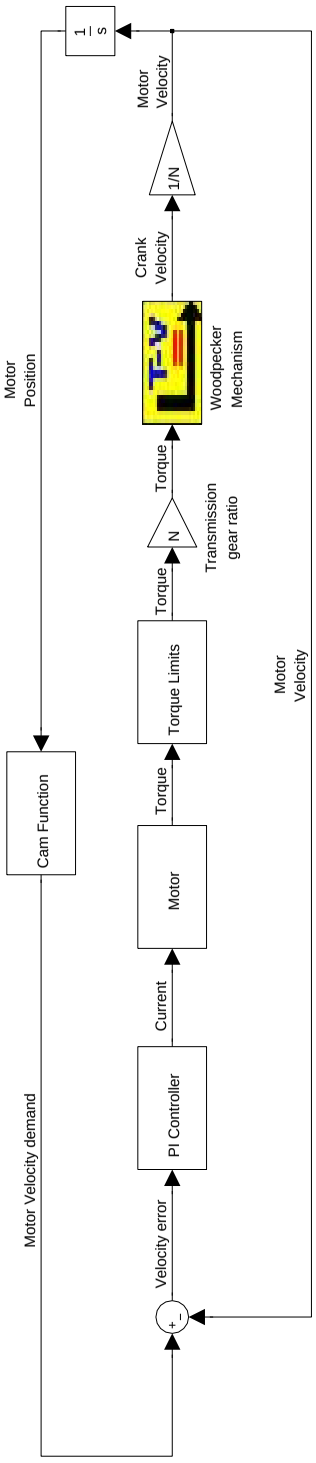


Figure 3.2: The implementation of the velocity cam function to the theoretical model of a test rig incorporating the Woodpecker mechanism

which produces a satisfactory velocity response. Analysis of the torque response of the system over a complete cycle should demonstrate torque response with a smaller peak-to-peak magnitude than that experienced using a constant velocity demand signal.

This process is shown diagrammatically in Fig. 3.3.

3.2 Simulating the Application of the Cam Function Generation Method to the Woodpecker Test Rig

The model of the Woodpecker test rig depicted in Fig. 2.21, was modified to include a torque truncation block mimicking the theoretical system discussed in the previous section. This modified model is depicted in Fig. 3.4.

The model was operated using both position proportional as well as velocity proportional and velocity integral gains. The gain values chosen match exactly those used when modelling the test rig in Ch. 2. As previously discussed in Sec. 2.2.4, the Deva004 controller could only operate in position control mode. This fact is reflected in Fig. 3.4, with the only input to the system being a position demand signal, P_{dem}. This being the case, the method of application of the Cam Function Generation Method had to be modified to accommodate this peculiarity of the Deva004. This was achieved in the following way:

1. To achieve Step 1 of the method described in Sec. 3.1, the velocity step demand signal of magnitude $V_{desired}$ had to be translated into the form of a time based position ramp signal. The steady state torque and velocity responses of the system were logged against torque truncation limits selected as before. Using these limits, the cam function was generated as before by passing the position ramp signal into the system. For the purposes of simulation, elapsed simulation

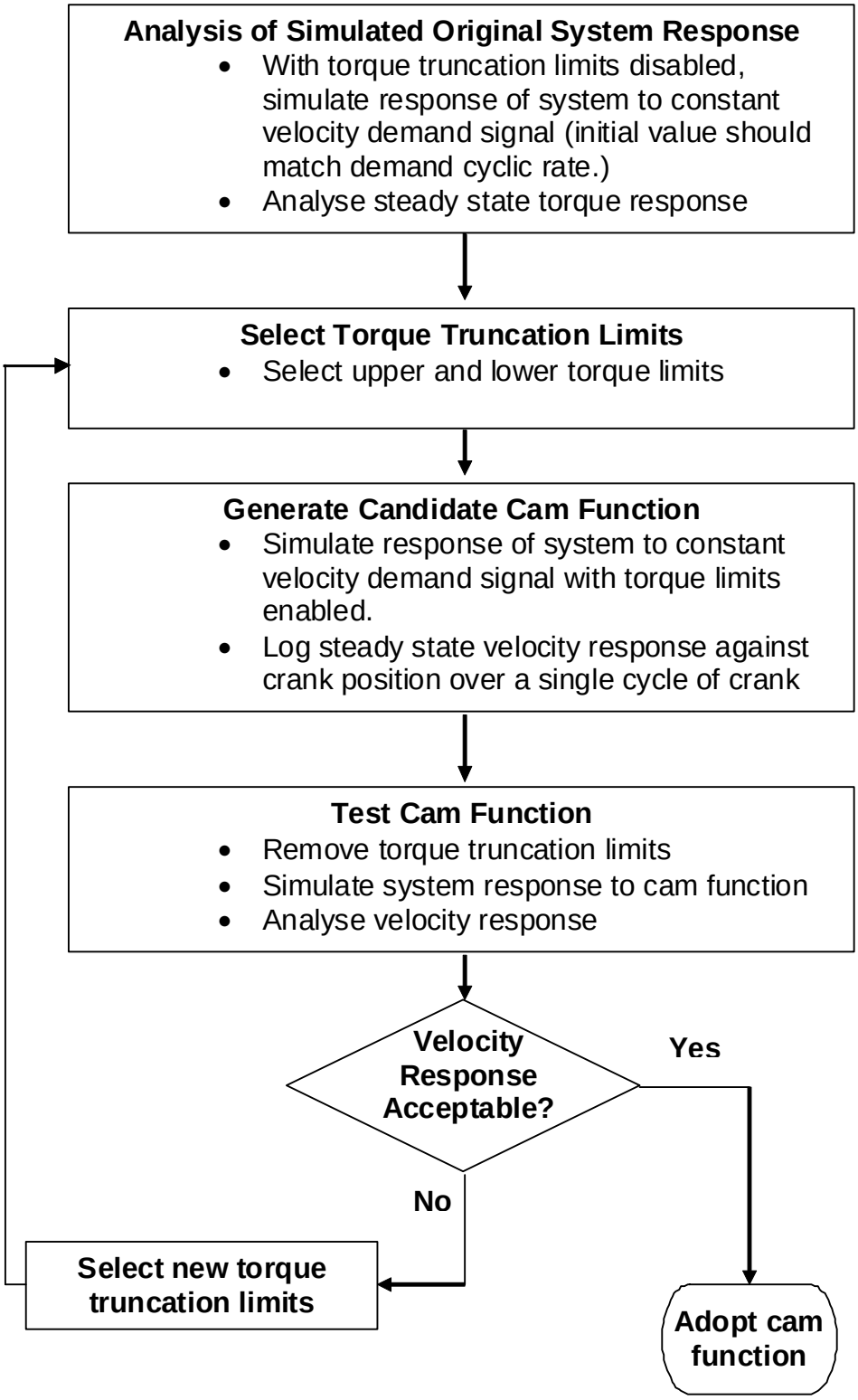


Figure 3.3: A flow chart of the Cam Function Generation Method

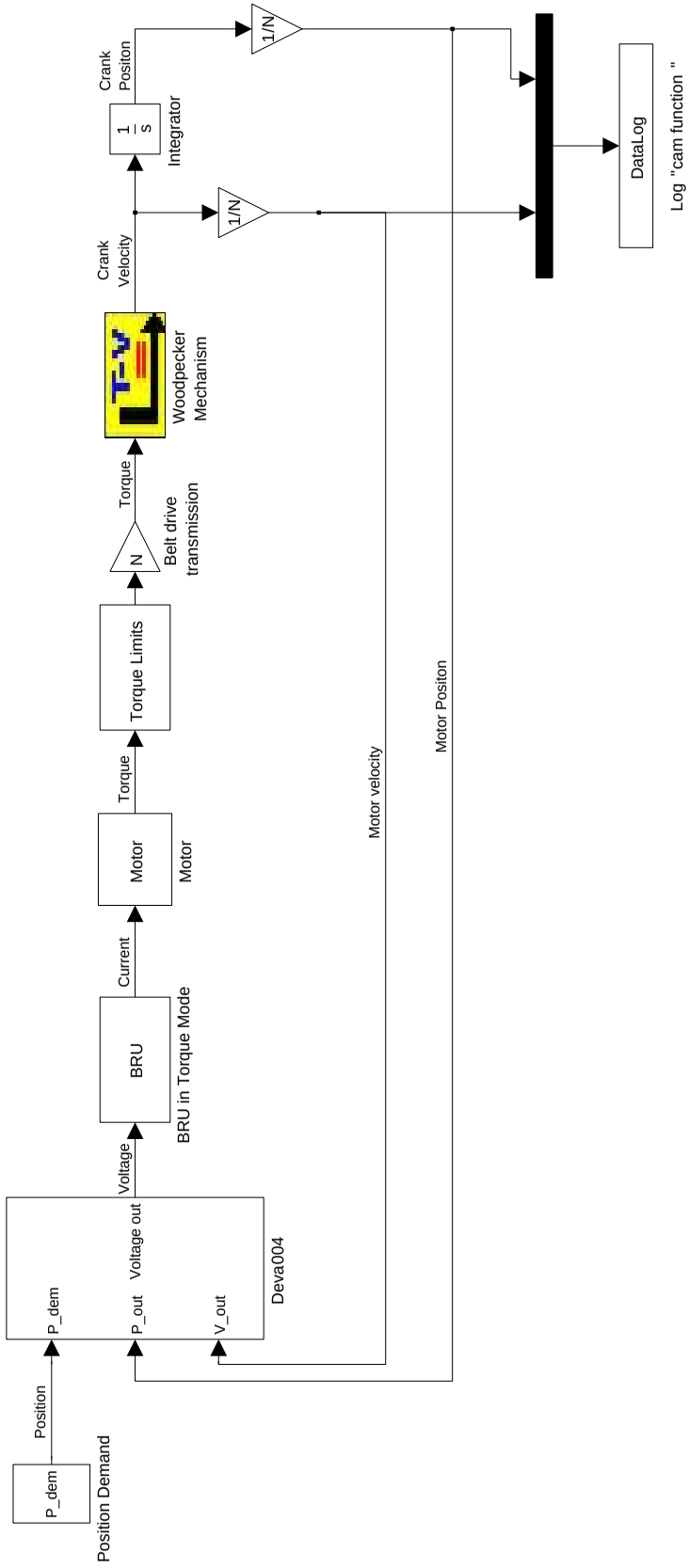


Figure 3.4: A model of the Woodpecker test rig with torque truncation introduced

time was also logged.

2. The cam function, consisting of both velocity and position data, was first considered logged against time. A script was written to mimic the internal processing logic of the Deva004, whereby a virtual motor was considered attempting to accelerate to each target velocity before achieving the corresponding target position, according to the rules defined in Sec. 2.2.5. The output of this script was a time based array, estimating the velocity profile of the internally generate velocity demand profile, V_{dem}, which the Deva004 would use to generate its motor command signal. Integrating V_{dem} generated a time dependent position demand profile. This profile was fed into the model of the test rig as the position demand signal in the model of the test rig depicted in Fig. 2.21.

3. In order to properly apply the cam function care had to be taken to ensure that the position of the crank at the instance when the cam function is invoked matches the initial position of the cam function. It must also be ensured that the velocity of the mechanism at the start of the crank cycle matches the initial velocity of the cam function, $V_{camInit}$. To achieve this, the gradient of the position ramp demand signal and thus magnitude of the constant velocity demand signal, V_{windup}, was varied. For each value of V_{windup} tested, the system was allowed to perform a sufficient number of cycles of crank, $n_{cycleWindup}$, to achieve steady state conditions. The simulated velocity of the mechanism crank at the end of the final cycle, $V_{windupEnd}$ was compared with $V_{camInit}$ and V_{windup} varied until the velocity of the mechanism at the end of final windup cycle, $V_{windupEnd}$, matches $V_{camInit}$. For the purposes of the simulation, the time taken for the mechanism to perform $n_{cycleWindup}$ cycles of crank with the demand signal V_{windup} was logged. This time was referred to as t_{windup}.

4. With torque limits removed, the model was configured to perform the following procedure:

   (a) Perform $n_{cycleWindup}$ complete cycles of crank with a constant demand velocity signal of magnitude V_{windup}. This was achieved by running the simulation

for the length of time t_{windup}.

(b) After t_{windup} has elapsed, invoke the controller response array generated in above, passing the array $\mathbf{P_{Cam}}$ into the controller as a time dependant position demand signal. The signal was repeated a number of times, $n_{cycleCam}$ to allow the system to operate under the influence of the cam function for several cycles.

The resultant velocity and torque responses of the system operating under the influence of the cam function were logged against crank position and compared with the initial response to the constant velocity demand signal as before. The output cyclic rate of the system, in response to the cam function, was compared to the desired cyclic rate, $V_{desired}$. If there was an unsatisfactory difference between the two values then the method was repeated with different torque truncation limits, as previously described, until the resultant output cyclic rate was satisfactorily close to $V_{desired}$.

Using this method, the model simulated the application of the method to generate motions with average cyclic rates of 200, 300, 450 and 600 rpm.

3.2.1 Results

Figure 3.5 shows a summary of the synthesised velocity reference cam functions used to generate cyclic motion with average cyclic crank velocities of 200, 300, 450 and 600 rpm. For ease of comparison, velocity values are shown in a normalised form as a proportion of the demand cyclic speed.

Each cam function generated a velocity response with an average cyclic rate satisfying the demand cyclic rate.

A summary of the constant velocity demand signal values and torque truncation limits used to generate the cam function, depicted in Fig. 3.5 is shown in Tab. 3.1.

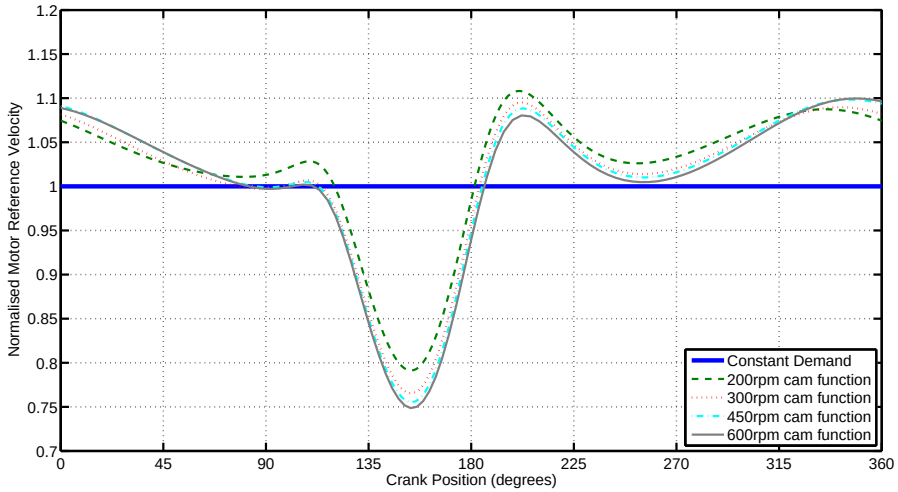


Figure 3.5: A comparison of velocity reference cam functions for operation at a variety of average cyclic crank velocities for use with the Woodpecker mechanism test rig. (Velocity values are normalised against demand average cyclic speed.)

	200 rpm	300 rpm	450 rpm	600 rpm
Upper Torque Limit (Nm)	2.0	2.97	4.46	5.95
Lower Torque Limit (Nm)	1.9	2.87	4.36	5.85
Windup Velocity Values ($v_{camInit}$) (rpm)	218.72	272.92	507.97	639.2

Table 3.1: The torque truncation limits used to generate the simulated cam functions

Correspondingly, Fig. 3.6 compares the torque responses of the system to the constant velocity and shaped cam function demands signals for the four operating speeds. Operating at cyclic crank velocities of 200, 300 and 600 rpm, reductions in torque variation peak-to-peak magnitude of 74.35%, 36.72% and 3.19% respectively, indicating that reductions in harmonic content amplitude should be experienced in the output motion. Operating at 450 rpm a torque variation peak-to-peak magnitude increase of 18.94% is experienced, indicating that more harmonic content may be experienced when operating at this speed. A summary of these variations in demand torque peak-to-peak magnitude is shown in Fig. 3.7.

It is noticeable that the method appears to be most effective when the mechanism is operated at 200 rpm. At faster speeds, the method becomes less effective. The reason for this discrepancy was attributed to the fact that the model of the test rig was derived using data obtained whilst the mechanism was actuated at 200 rpm, making

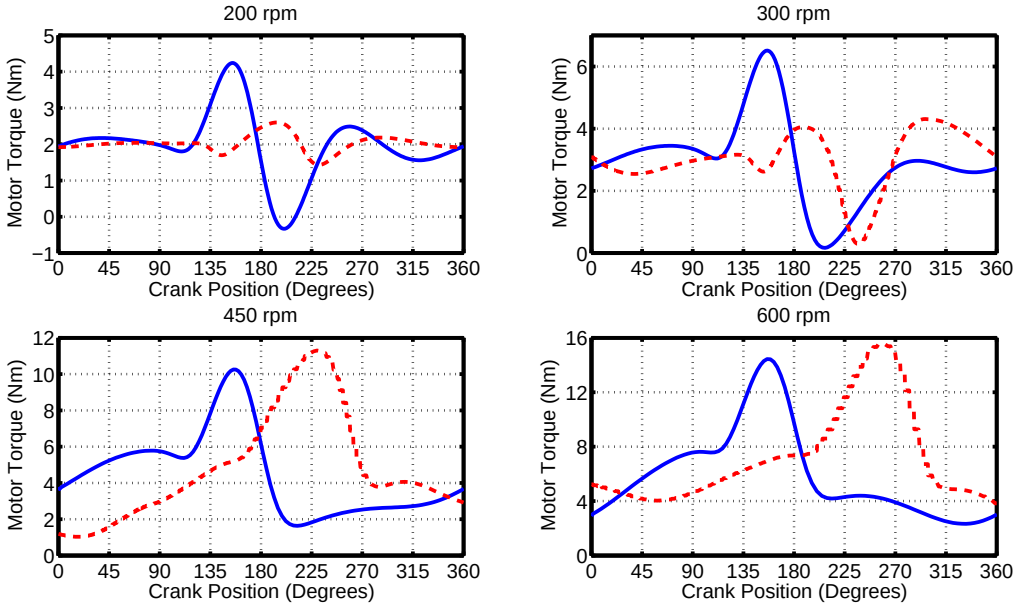


Figure 3.6: A comparison of Woodpecker mechanism test rig simulated torque responses to a variety of velocity cam functions for operation at different average cyclic crank velocities. Solid lines correspond to responses to constant velocity demands signals and dashed lines correspond to responses to cam functions.

the model most accurate, simulating the motion of the system operating at this speed. At speeds significantly different from this operating point, it is entirely possible that system parameters may change drastically, making the model inaccurate. Evidence for such behaviour was demonstrated in Ch. 2, whereby the controller gain, K_m,varied significantly as motor payload and operating speed varied. It is possible that when actuating at this speed, resonance is achieved in the system. Due to the physical limitations of the mechanism, it was not possible to test either of these hypotheses. Actuating the test rig at crank velocities well in excess of 200 rpm was deemed unsafe, as it would risk overloading the mechanism, resulting in structural failure.

As implied in the literature, reducing the peak-to-peak magnitude of the variation in torque that the drive motor must exert on a system to generate a given motion will result in a reduction in the amount of harmonic content in the output motion [27, 31]. Work was carried out to ascertain the effectiveness of the cam function in reducing harmonic content in the output motion of the system. The output motion of the system, in response to a 200rpm constant velocity demand signal, was compared to the

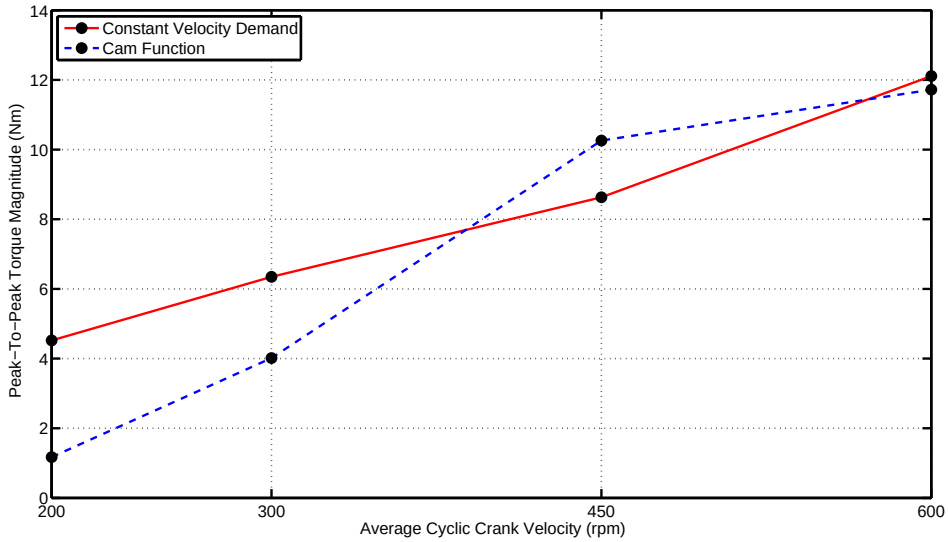


Figure 3.7: A comparison of the peak-to-peak magnitude values of the simulated torque responses of the Woodpecker mechanism test rig in response to velocity reference cam functions for operation at different average cycle crank velocities

equivalent response to the 200rpm cam function. As carried out by Yuan et. al. [27], the output angular motion of the link bearing the end effector, link 7, was analysed for vibrational content with respect to the input angular motion of the crank, link 1. The relevant data was processed using Fourier analysis [62]. Using this method, the data was broken down into its constituent frequencies, allowing the makeup of the harmonic content present in the output motion to be seen. The results of this analysis can be seen in Fig. 3.8.

Analysis of Fig. 3.8 shows that the amplitude of the harmonic content present in the output motion generated using the cam function is consistently smaller than the equivalent harmonic content present in the output motion generated using the constant velocity demand signal. The difference in amplitude is more significant at the higher frequencies. A comparison of the high frequency responses is shown in Fig. 3.9.

This result indicates that the use of a cam function would indeed reduce the harmonic content in the output of the mechanism. It also shows that reducing the peak-to-peak magnitude of the variation in propulsive torque reduces results in a reduction in the

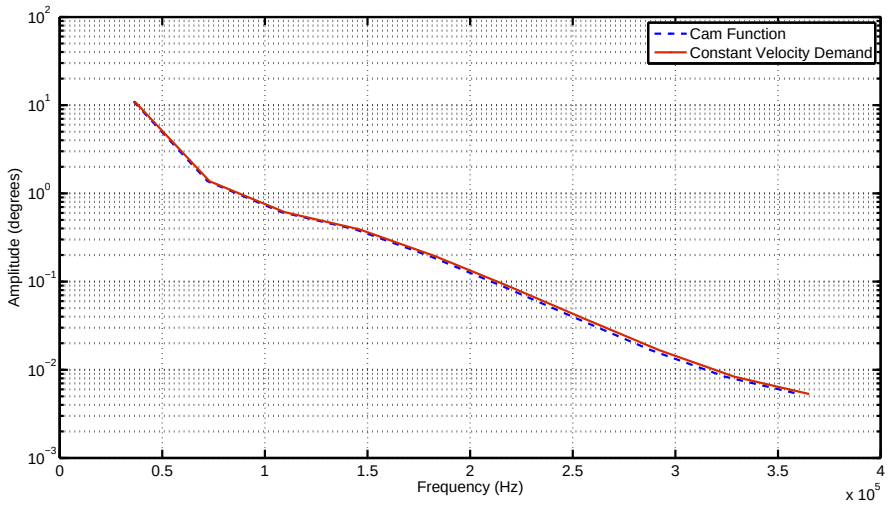


Figure 3.8: A comparison of the harmonic content in the output motion of the Woodpecker mechanism in response to a 200 rpm constant velocity demand signal and in response to cam function generating a motion of 200 rpm.

magnitude of the harmonic content in the resultant output motion confirming the link implied in the literature.

In terms of practical application of the method, the great strength of the Cam Function Generation Method is the fact that benefits can be achieved without making any modifications to the system. Instead the dynamic benefits are achieved through the use of a new variable velocity demand signal. Thus assuming a good model of the system under scrutiny can be obtained, implementation of the method is easy to achieve taking the form of a simple software update. This method is therefore particularly suitable for the improvement of the dynamics of existing mechanisms.

With such encouraging results being obtained from the 200 rpm cam function response, it was decided that the method appeared promising and merited further work.

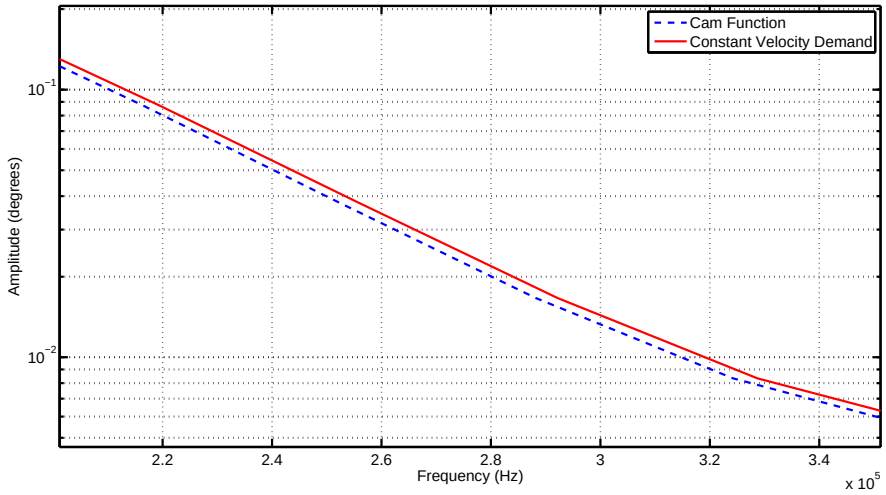


Figure 3.9: A comparison of the high frequency harmonic content in the output motion of the Woodpecker mechanism in response to a 200rpm constant velocity demand signal and in response to cam function generating a motion of 200 rpm.

3.3 Experimental Validation of the Cam Function Generation Method

Previously in this chapter, work has been discussed to develop cam functions to actuate the Woodpecker test rig at a variety of cyclic rates. These cam functions have been tested using simulation work. The simulations showed that the biggest gain in dynamic performance could be achieved using the 200 rpm (crank velocity) cam function. To verify this result, work was carried out to apply the cam function to the real world test rig.

3.3.1 Applying the Cam Function to the Test Rig

The 200 rpm (crank velocity) cam function, depicted in Fig. 3.5, was discretised by the simulation program and exported from the simulation program in the form of two ASCII files for use with the Deva004. One file contained a list of discrete target velocities, whilst the other contained a list of corresponding target positions. The cam function was sampled with a frequency of 50Hz. Investigative work had found that the Deva004

would malfunction if the data was sampled any higher due to its inability to handle more many precision points. Care was taken to ensure that the sampled cam function still contained sufficient data to properly describe the magnitudes and form of cam function. To ensure this was achieved, a number of additional points were manually added to the ASCII files.

A control script was written to command Deva004 to read both files and when invoked, to command the motor to move to perform the cam function. In so doing, the controller would aim to achieve each target position with a final velocity equal to the corresponding target velocity value.

Care was taken to ensure that the position of the mechanism crank at the instance when the cam function was invoked matched the initial position of the cam function. It was also necessary to ensure that the velocity of the mechanism crank at the end this position matched the initial velocity of the cam function, $V_{camInit}$. To achieve this, the mechanism was actuated with a constant velocity demand signal of magnitude $V_{windupReal}$ and allowed to perform $n_{cycleWindupReal}$ cycles to allow sufficient time for the system to achieve steady state conditions. The final velocity of the mechanism at the end of the cycle, $V_{windupEnd}$, was compared to the starting velocity of the cam function, $V_{camInit}$, and $V_{windupReal}$ adjusted until $V_{windupEnd}$ was close to $V_{camInit}$. When iterating values of $V_{windupReal}$, care was taken to ensure that the mechanism was operating within its stable and safe region of operation.

With all preparatory work done, a script was written to apply the cam function to the test rig using the following method:

1. Starting from rest, the mechanism was actuated for $n_{cycleWindupReal}$ cycles of crank using a constant velocity demand signal of magnitude $V_{windupReal}$.
2. The cam function was then invoked. The mechanism performed $n_{cycleCam}$ cycles of crank under the control of the cam function.
3. The mechanism ran for a further $n_{cycleWindupReal}$ cycles of crank, before being

decelerated to rest.

The response of the system throughout the procedure was logged, as described in Sec. 2.3.1.

The response of the system under control of cam function was compared to the response of the system to the constant velocity demand signal of magnitude $V_{desired}$.

3.3.2 Results

Figure 3.10 compares the resultant velocity command output response of Deva004 motion controller, V_{dem}, as output to the motor in response to the sampled cam function as generated by the simulation program, with the original unsampled cam function. It is the signal V_{dem} which the controller uses to generate command signals for the motor.

The mechanism wound up to speed with a constant velocity demand speed of 215 rpm.

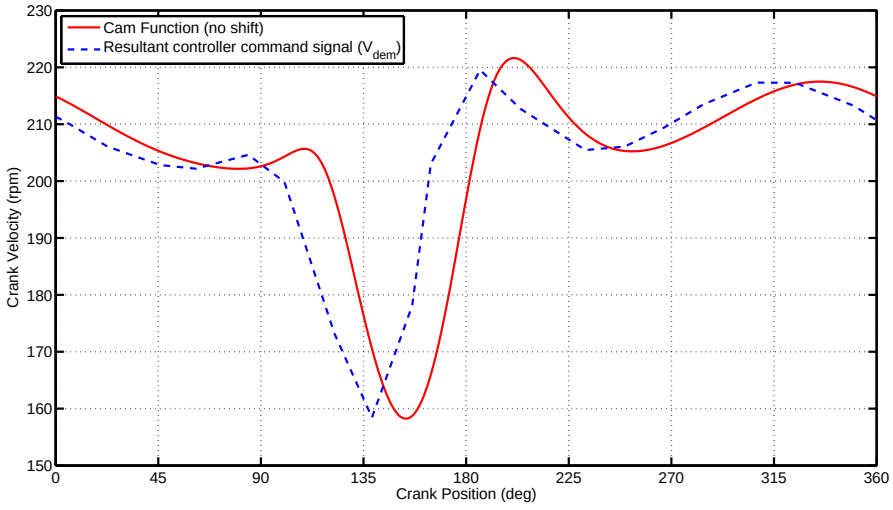


Figure 3.10: A comparison of the experimentally derived steady state velocity responses of the Woodpecker mechanism test rig in response to a velocity cam function for operation at an average cyclic crank velocity of 200 rpm and a velocity step demand signal of 200 rpm (crank speed) ($Kp_{pos} = 100$, $Kp_{vel} = 200$ and $Ki_{vel} = 400$)

It can be seen from Fig. 3.10 that through the process of signal processing by the

controller, a phase shift of approximately 15 degrees of crank is introduced between the cam function, as specified in the input data files and the velocity command signal to the motor, V_{dem}. Consequently, the shaped demand signal sent to the motor is out of phase with the dynamics of the mechanism, potentially limiting the effectiveness of the method. The response of the test rig to V_{dem} in this form is shown in Fig. 3.11.

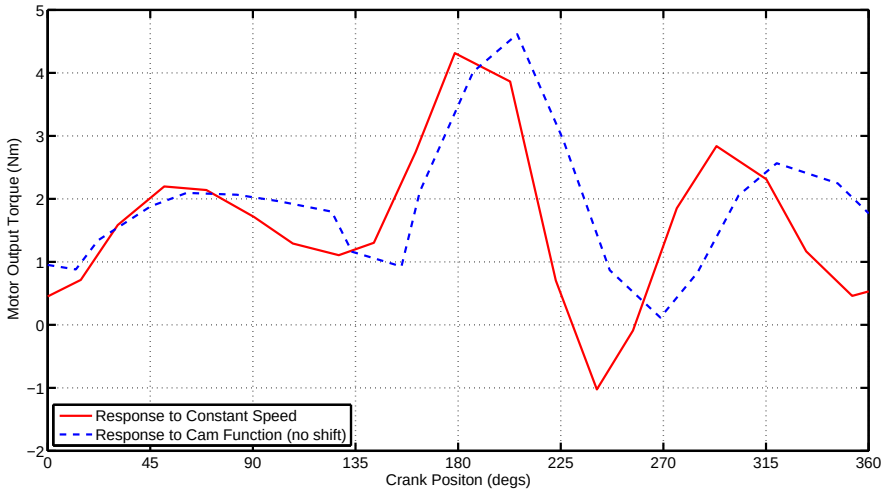


Figure 3.11: A comparison of the experimentally derived steady state torque responses of the Woodpecker mechanism test rig in response to a velocity cam function for operation at an average cyclic motor velocity of 200 rpm and a velocity step demand signal of 200 rpm (crank speed) ($Kp_{pos} = 100$, $Kp_{vel} = 200$ and $Ki_{vel} = 400$)

From Fig. 3.11 it can be seen that when compared to the equivalent response to the constant velocity demand signal, the cam function applied in this manner reduces the magnitude of the peak-to-peak torque response from 5.6 Nm to 4.49 Nm. This signifies a reduction of 1.11 Nm or 21%.

To compensate for this phase shift the cam function was modified. The cam function was progressed forwards by 15 degrees of crank to counter the shift. As expected, passing this new function into the controller generated a resultant V_{dem} signal, which was now in phase with the unsampled cam function and thus in phase with dynamics of the mechanism. This is shown in Fig. 3.13.

The resultant torque response to this new phase corrected cam function shows a more significant reduction in peak-to-peak torque variation magnitude with these values

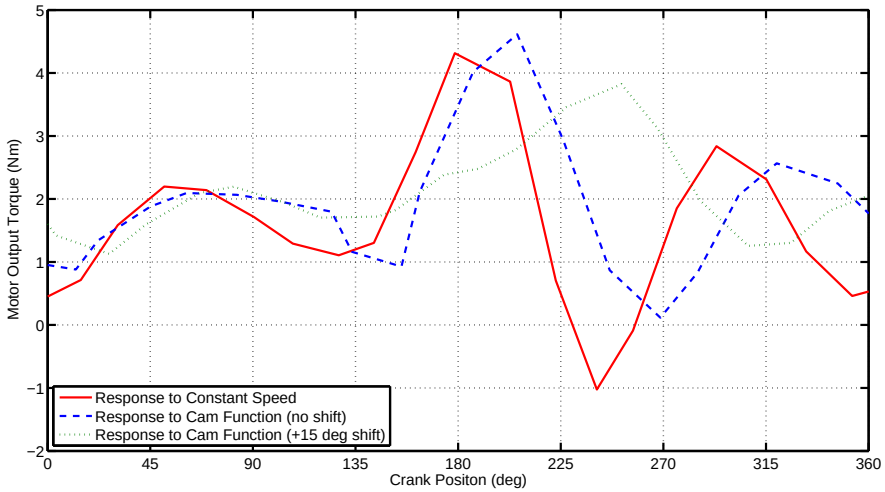


Figure 3.12: A comparison of the experimentally derived steady state torque responses of the Woodpecker mechanism test rig in response to a velocity cam function for operation at an average cyclic crank velocity of 200 rpm a the same cam function with phase correction ($Kp_{pos} = 100$, $Kp_{vel} = 200$ and $Ki_{vel} = 400$)

being reduced from 5.6 Nm to 2.1 Nm, a reduction of 3.5 Nm or 63%.

The velocity response of the system to the phase corrected cam function is shown in Fig. 3.13.

The corresponding velocity response to the cam function is also satisfactory. The resultant steady state cyclic rate achieved is approximately 203 rpm, slightly faster than desired 200 rpm. Despite there being no facilities for measuring the amplitude of the harmonic content present in the output motion of the test rig, it was observed that the test rig was audibly quieter when responding to the cam function than to the constant velocity demand signal.

3.4 Conclusions

The work carried out in this chapter demonstrates the benefit of using shaped velocity cam functions as velocity demand signals in place of constant velocity demand signals to achieve reductions in magnitude of peak-to-peak torque responses resulting in output

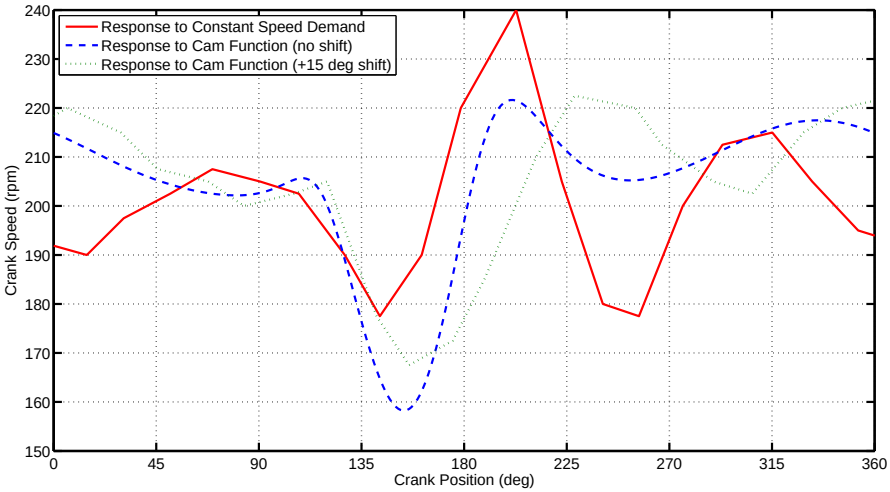


Figure 3.13: A comparison of the experimentally derived steady state velocity responses of the Woodpecker mechanism test rig in response to a phase corrected velocity cam function for operation at an average cyclic motor velocity of 200 rpm and a velocity step demand signal of 200 rpm (crank speed) ($Kp_{pos} = 100$, $Kp_{vel} = 200$ and $Ki_{vel} = 400$)

motions with reduced harmonic content. The work in this chapter utilised a pre-derived model of the Woodpecker test rig, modelled in Ch. 2. Using this model, a method was developed to synthesise a position dependent variable velocity demand signal or *cam function*. This signal was sympathetic to the dynamics of the test rig and mechanism.

The model was used to simulate the generation and application of a series of cam functions, which would actuate the test rig at various cyclic rates ranging from 200 rpm up to 600 rpm. When compared to simulated responses to equivalent constant velocity demand signals simulations demonstrated reductions in the peak-to-peak magnitude of actuation torque of as much 74.35% when operating at 200 rpm crank velocity. At higher velocities, the method was less effective. The reason for this behaviour was attributed to the model of the test rig using system parameters obtained using experimental data obtained by cycling the mechanism at 200 rpm. The model was therefore tuned for operation at this speed. At speeds significantly different from this, it was hypothesised that the identified co-efficient of friction, b_{comb}, and system gain value, K_m, were unrealistic for operation at these speeds.

Spectral analysis was carried out to compare the harmonic content in the output motion

of the system in response to a 200rpm constant velocity demand signal and with cam function inducing a motion with an average cyclic speed of 200rpm. The spectral analysis showed that using the cam function in place of the constant velocity demand signal resulted in a decrease in harmonic content amplitude, particularly at high frequencies.

With such encouraging simulated results obtained for operation at 200 rpm, it was decided that the method was promising and worthy of further work. Based on this result, work was done to apply the method to the actual, real world prototype servomechanical test rig which had previously been modelled. A cam function was developed and applied to the Woodpecker test rig, which simulations showed should actuate the rig at an average cyclic rate of 200 rpm. In response to the cam function, the peak-to-peak magnitude of the resultant torque variation was 63% smaller than the equivalent response to the equivalent constant demand velocity signal. This demonstrated the effectiveness of the method. Although the test rig was equipped with no means of measuring the amplitude of any induced harmonic content,it was observed that the test rig was quieter when operating using the cam function than when operating using the constant velocity demand signal indicating a reduction in harmonic content.

When applying the method to the Woodpecker test rig, the cam function had to be shifted by 15 degrees of crank to compensate for a phase shift between the cam function and the command signal sent to the motor. This phase shift was apparently introduced by the internal signal processing of cam function by the motor controller.

It was identified that the great strength of the method described in this chapter lies in the fact that minimal modification to the system is needed to implement it. The only modification to the system that is necessary is the redefinition of the demand signal. The method is therefore cheap, quick and easy to implement.

3.5 Future Work

The work described in this chapter detailed a method of deriving velocity cam functions. To derive these cam functions, simulations were performed, in which the user specified upper and lower torque truncation limits. The specification of the values of these limits was performed manually, using an iterative method. A logical development of the Cam Function Generation Method would be to implement a method of automating the process of selecting these upper and lower torque limits. This could be done by considering the problem as a multi-parametric numerical optimisation problem. A flow diagram of such a process is shown in Fig. 3.14

From Fig. 3.14, it can be seen that this process acts to vary the upper and lower torque limits, in order to minimise the overall cost value R_T. R_T is the sum of weighted velocity and torque cost values, R_{Vel} and R_{Tq} respectively. The lower the values of these cost parameters, the better.

With candidate torque limits specified, the process generates a candidate cam function and analyses the cyclic velocity and torque responses of the test rig to the resultant cam function. The velocity cost function would be derived from analysis of the cyclic velocity response. The function for this cost value should act to generate increasingly large values of cost, the more the resultant cyclic rate deviates from the demand cyclic rate. Correspondingly, the torque cost function would be derived from analysis of the peak-to-peak magnitude of the cyclic torque response. The cost function should be specified such that cost values increase with peak-to-peak magnitude. In this way, the optimisation process will seek to generate cam functions which generate motion with cyclic rates close to the demand cyclic rates and which also have torque responses with minimal peak-to-peak torque magnitude.

The use of an optimisation routine to specify upper and lower torque limits has a number of advantages. Firstly, it automates the process of torque limit selection removing the necessity for manual intervention. Secondly, it could result in the generation of

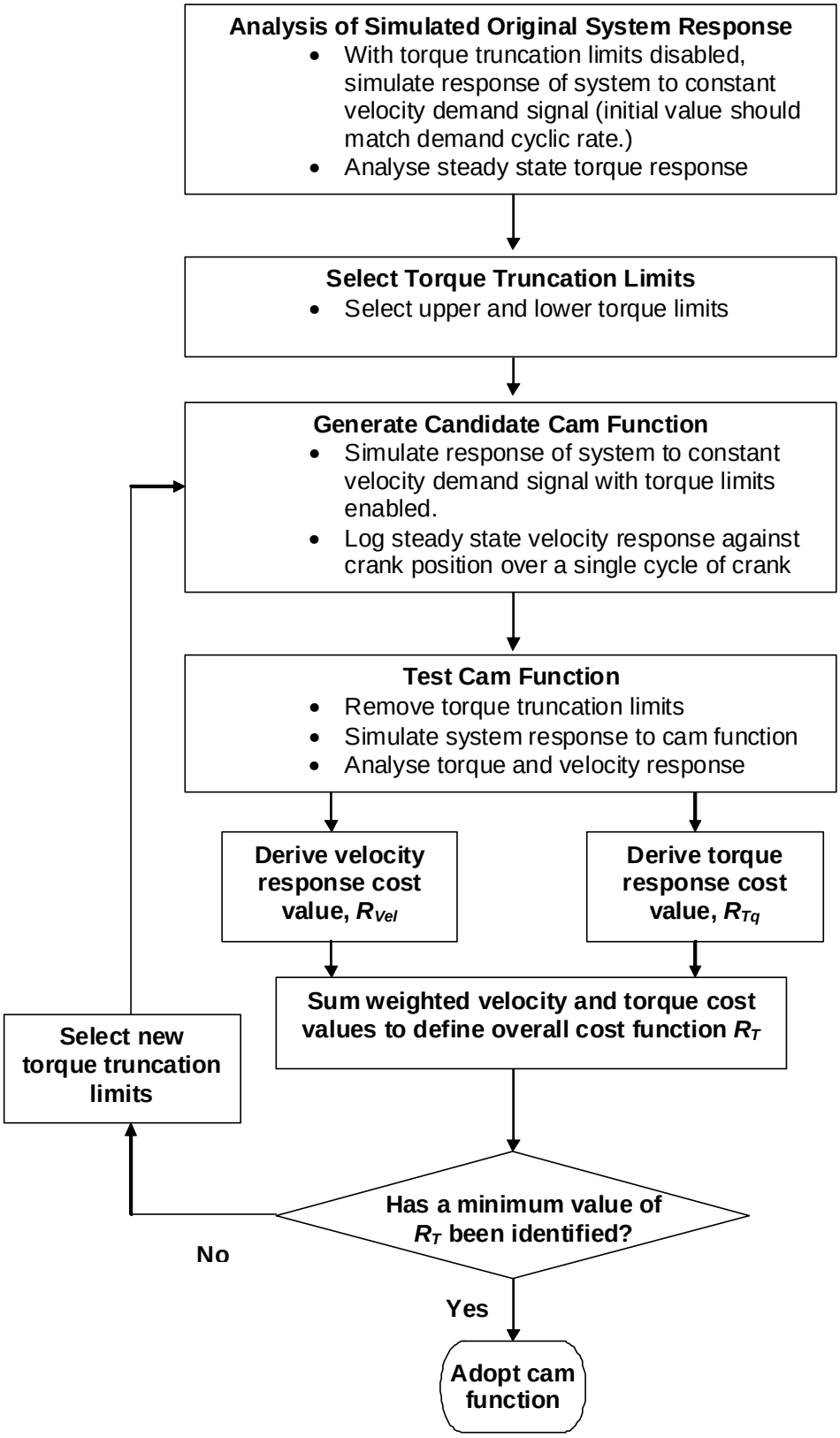


Figure 3.14: Automating the selection of torque truncation limits in the Cam Function Generation Method

more effective cam functions since the search would be more exhaustive than the manual iterative process described in this chapter.

It is important to note that work discussed in this chapter aims to create cam function, which reduce the resultant peak-to-peak torque variation magnitude of the system. The method could equally be applied using an alternative metric as a measure of success. So for example, the method could seek to reduce cyclic energy consumption of the system. In this way the method can be seen to be not only effective but also versatile.

Chapter 4

Combining Traditional Mechanism Synthesis Methods With Inverse Dynamics

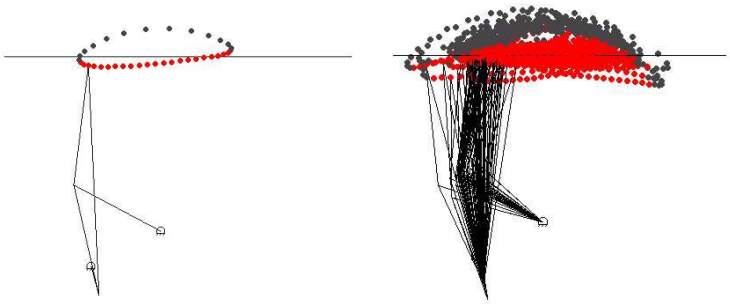
The problem of designing a mechanism for high speed operation can be seen to comprise of two independent but potentially conflicting design tasks, namely kinematic accuracy and dynamic performance. As discussed in Sec. 1.2 the majority of traditional mechanism synthesis methods consider only kinematic, path following performance, with dynamic performance rarely given much consideration. This approach can lead to the generation of mechanisms which can follow a desired output path with great accuracy but exhibit poor vibrations characteristics when actuated at high speeds.

Work described in this chapter will discuss the development of a unique, computer based, multi-stage mechanism design strategy which combines the kinematic analysis abilities of a traditional mechanism synthesis tool with a dynamic analysis method, to synthesise a mechanism with high quality kinematic and dynamic characteristics. The approach will first be presented before it is applied to the Woodpecker mechanism design. It is important to note that the aim of the work described in this chapter is

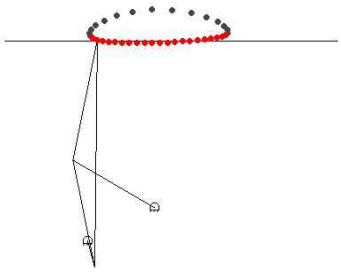
not to develop a robust mechanism synthesis method. Instead, the work will seek to demonstrate, through the use of a mechanism design strategy, the potential benefits of considering mechanism dynamics and kinematics when designing a mechanism.

The design of mechanisms and the development of approaches to support their effective design, can be traced back to the days of Leonardo DaVinci [63]. Early researchers would use simplified scale models that would be incrementally refined to achieve the desired motion. Such techniques remained common practise until the development of early Computer Aided Design (CAD) systems, with which it was possible to incrementally change the geometry of computer generated models of mechanism links and to visualise the output motion of the resultant mechanism. Such techniques, although manual, resulted in significant design improvements and enabled the design envelope for given mechanisms to be explored more rapidly and in more depth. Since these early CAD systems, the ability to simulate motion has become common place in most Computer Aided Engineering (CAE) systems [64–66]. In addition to the functionality offered by these general CAE systems, there are a variety of computational approaches that have been created with the specific aim of supporting the design and analysis of mechanisms [67, 68]. These include the use of parametric models and optimisation methods to computationally explore the design space and best-achieve the desired motion requirements. Figure 4.1 shows an example of the process of parametric design and optimisation.

To initialise this design process, the user must first define the desired output path of the final mechanism. The style of the mechanism, which is to be optimised must also be defined. This is equivalent to *number synthesis*, as discussed in Sec. 1.2. In the depicted example, a five bar linkage mechanism is being considered. An initial guess at suitable link dimensions must also be made by the user. Figure 4.1(a) shows a seed mechanism and its output motion denoted as series of discrete points. The feature of this output motion, which is most essential for operation, is the ability of the mechanism to follow a straight, horizontal path. The horizontal path is depicted by a solid black line. The form of the remainder of the output path is unimportant. Here, the design task is to



(a) Seed Mechanism (b) Design alternatives



(c) Final mechanism design

Figure 4.1: Parametric design and kinematic optimisation of a mechanism

perform *dimension synthesis* to determine the values of link lengths which can best achieve the desired output path. To do this, a parametric model of the mechanism is constructed and an objective cost function is specified, which takes into consideration the deviation of the output motion of the mechanism from the desired path. In this example, only the lower (red) portion of the motion is considered. Within the modelling environment an optimisation function is then invoked [67] and the design parameters varied generating a variety of candidate mechanism designs, as shown in Fig. 4.1(b). After a predefined number of iterations, or when an optimal solution is determined, the final mechanism design is captured, as shown in Fig4.1(c).

4.1 Combining Dynamic and Kinematic Analysis

In this section, a unique, computer based mechanism design strategy will be described, which combines a traditional, kinematic mechanism synthesis method with a dynamic analysis procedure. Starting with a seed mechanism and a desired output path, the

method aims to perform *dimension synthesis* to identify a mechanism design with improved dynamic and kinematic characteristics. A flow chart of this strategy is shown in Figures 4.2 and 4.3.

4.1.1 The Definition of a Seed Mechanism

As a starting point for this strategy, a desired output path must be defined. *Number synthesis* must then be carried out to define a seed mechanism. This should be done with reference to the desired output path and can be achieved using conventional mechanism design tools. Sensible estimates of suitable mechanism link dimensions should be made. These values will act as the initial conditions for the *dimension synthesis* stage of the method. For the purposes of dynamic analysis, theoretical mass and inertia scaling factors for each link should also be estimated.

To apply the mechanism synthesis strategy, a non-critical portion of the output path must exist in the desired output path. In other words, it should be possible to alter the form of the desired output path whilst still fulfilling its functional purpose. For example the recoil portion of the motion, where the mechanism is passing through free space could be described as being *non-critical*, since the exact path followed by the mechanism can take any form. Conversely the portion of the output path where the mechanism interfaces with a product could be *critical*, since the form of the output motion of the mechanism in this portion of the motion is fundamental to its functionality. The critical and non-critical portions of the desired output path must be clearly defined.

4.1.2 Mechanism Synthesis and Kinematic Analysis

The purpose of defining the critical and non-critical portion of the desired output path is to allow the synthesis method scope to vary the form of the non-critical portion of the desired output path. Dimension synthesis can then be carried out using a variety of alternative output paths, to identify corresponding mechanism designs, which may

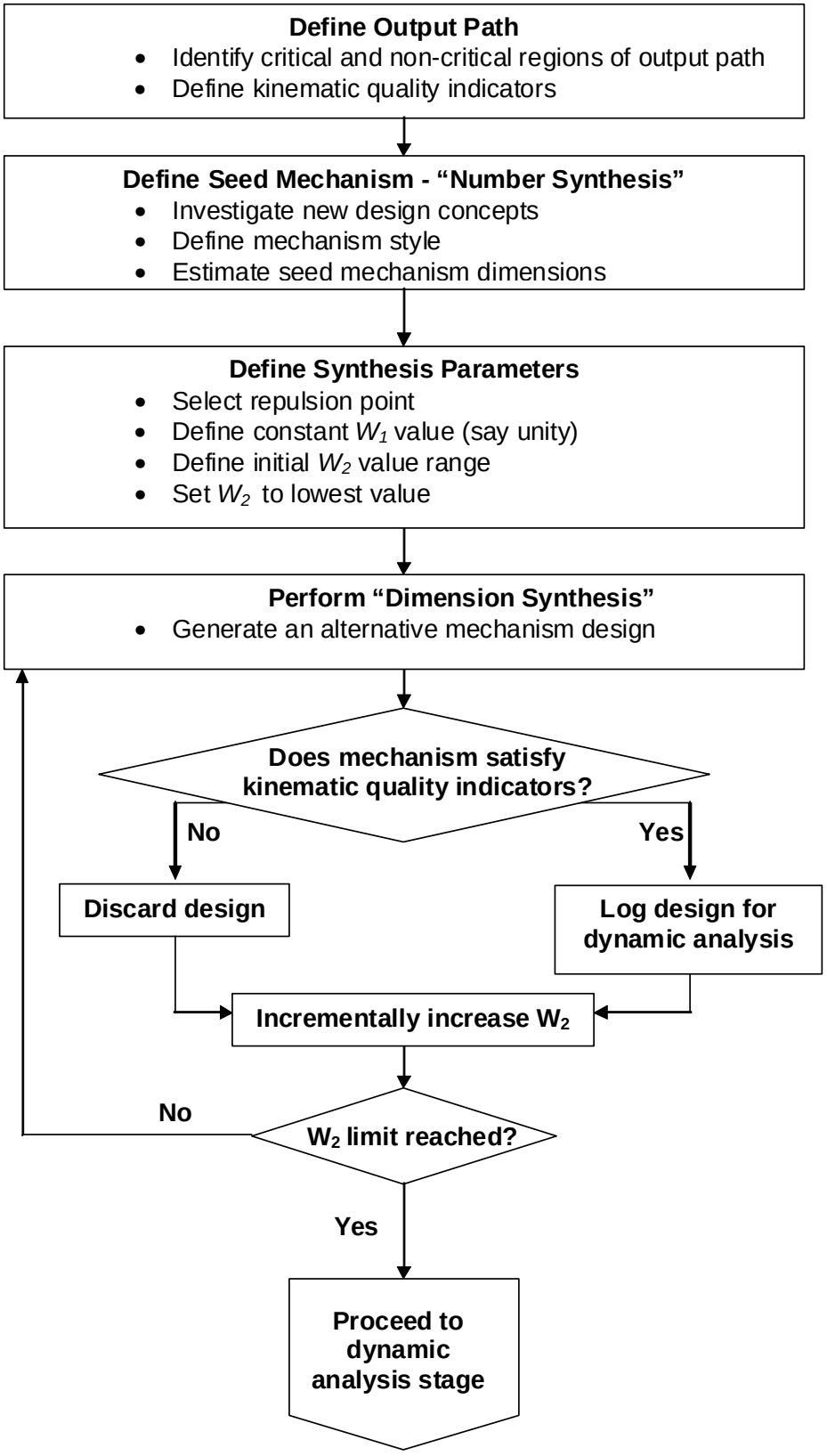


Figure 4.2: The kinematic analysis stage of the combined mechanism synthesis process and dynamic analysis process

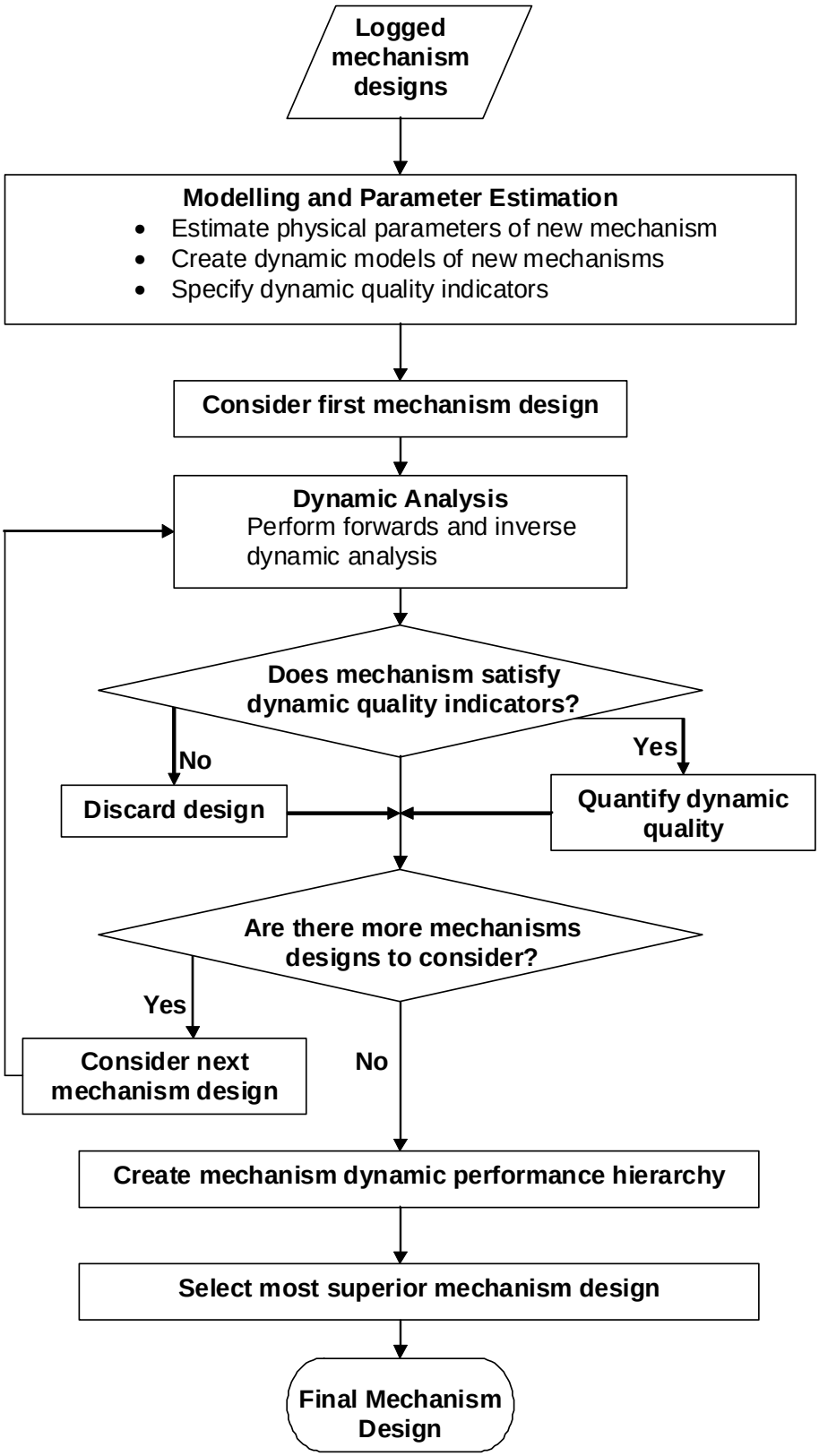


Figure 4.3: The dynamic analysis stage of the combined mechanism synthesis process and dynamic analysis process

possess optimal dynamic performance characteristics. Starting with the initial desired path, a method was devised to force the synthesis method to explore different paths.

The output path must first be expressed as a series of discrete points. A single point on the non-critical portion of the motion should then be selected. The other points on the non-critical portion of the path are discarded. The *dimension synthesis* objective cost parameter to be minimised, R_K, can be expressed as follows.

$$R_K = W_1 R_1 + W_2 R_2^{-1} \tag{4.1}$$

For each candidate mechanism, the synthesis program estimates the corresponding output path as a series of discrete points. The method then calculates the shortest distance between each point of the critical portion of the desired output path and the a point on the synthesised output path. The sum of the distances is the parameter R_1. The same method is performed for the repulsion point. This distance is the parameter R_2. W_1 and W_2 are weighting factors, which influence the amount of cost significance given to the critical portion of the path and the repulsion point respectively. Due to the R_2^{-1} term in the cost function, any synthesised path passing near the repulsion point will result in large values of cost. Hence, as the ratio between W_2 to W_1 is increased, the *repulsiveness* of the repulsion point will increase, resulting in the synthesis process seeking to generate mechanism designs with output paths that pass ever further away from the repulsion point. By iteratively varying W_2 and keeping W_1 constant, a wide variety of alternative mechanism designs can be obtained.

The *dimension synthesis* portion of the method yields a variety of alternative mechanism designs, some of which may be unsuitable for the desired application. To discriminate between suitable and unsuitable mechanism designs, it was necessary to devise a method of analysing each design for kinematic quality. To achieve this, a series of kinematic quality indicators need to be agreed upon. Such indicators may include mechanism size, output path accuracy and so on. Only those mechanisms which satisfy the kinematic quality indicators are considered for dynamic analysis.

4.1.3 Modelling and Parameter Estimation

In order to analyse the dynamics of the varying mechanism designs, it is necessary to quickly estimate mass and inertia values for each link. For a given mechanism link, n, mass and inertia scaling factors, m_{0n} and J_{0n} correspondingly, need to be specified. A suggested method of deriving these values is to estimate the mass and inertia values, m_n and J_n for each link of the Seed mechanism. This can be done by modelling a typical link with length l_n, in a CAD package. The mass and inertia scaling factors, m_n and J_n respectively, can then be estimated as follows:

$$m_{0n} = \frac{m_n}{l_n} \tag{4.2}$$

$$J_{0n} = \frac{J_n}{m_n l_n^2} \tag{4.3}$$

Using m_{0n} and J_{0n} as scaling factors, mass and inertia values can be estimated for each corresponding link for the new, synthesised mechanisms. To calculate the mass of link n with length l_n, the mass m_n of the new link can be calculated as:

$$m_n = m_{0n} l_n \tag{4.4}$$

Similarly, the inertia value J_n of link n can be calculated as:

$$J_n = J_{0n} m_n l_n^2 \tag{4.5}$$

The links of the synthesised mechanisms were considered to have their centres of mass at their mid-point. The inertia of the triangular tertiary link was obtained by summing the inertias of the constituent links. Similarly, its mass was estimated by summing the masses of the constituent links. Its centre of mass could then be calculated directly.

4.1.4 Dynamic Analysis

Using DYSIM, it is possible to create dynamic models of each candidate mechanism and analyse and scrutinise each design for dynamic performance using forward and inverse dynamics. Using these models, simulations can be run to estimate the dynamic behaviour of a mechanism design and to analyse it for specific dynamic performance requirements. The dynamic quality of a given mechanism design can be quantified using a variety of indicators, such as driving torque variation peak-to-peak magnitude or perhaps cyclic energy consumption. The most desirable mechanism can then be identified.

4.2 Applying the Method to The Prototype

To test the method described in this chapter, the *Woodpecker mechanism*, as described in Ch. 2, was considered. The design of this mechanism is shown in Fig. 4.4. This design was adopted as the seed mechanism. It is known that seed mechanism can follow the output path. Using mass and inertia data taken from the CAD model of the mechanism, mass and inertia scaling values, m_{0n} and J_{0n} respectively, for each link were estimated. This data was used to determine m_{0n} and J_{0n} values.

The discretised output path of the seed mechanism is shown in Fig. 4.5. This orbit will be adopted as the original desired output path. The end effector orbits the path in a clockwise direction.

Figure. 4.5 also shows the discrimination between the geometrically critical and non-critical portions of the path. The upper portion of the motion is geometrically critical, as it is in this region that the mechanism interfaces with the product. In the non-critical portion of the motion, the mechanism recoils through free space. The path which the mechanism follows in this region is therefore non-critical. A kinematic quality indicator was defined, stipulating that a satisfactory mechanism design should have an output

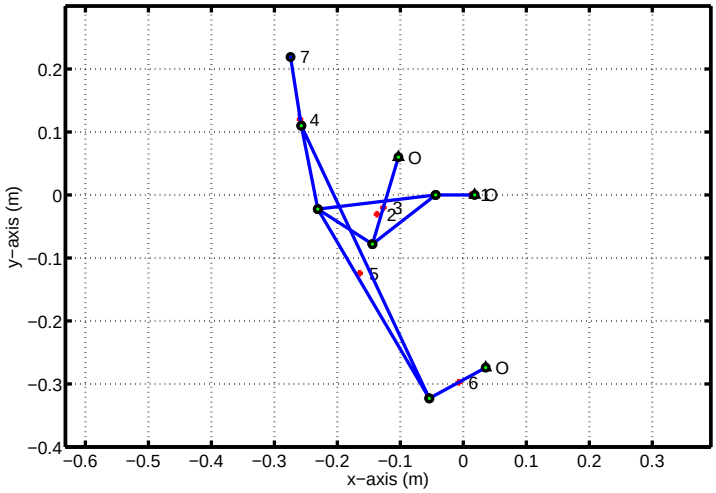


Figure 4.4: The Woodpecker mechanism (seed mechanism)

path with minimal vertical deviation in the critical portion of the motion.

For the purpose of generating suitable alternative mechanisms, it was decided that it may be possible to achieve dynamically superior mechanism designs by reducing the vertical height of the output path. To achieve this, a repulsion point positioned just below the lowest point on the output path was selected.

A dynamic quality indicator was specified to try to minimise the peak-to-peak magnitude of the torque demand needed to actuate each mechanism at a given speed. This indicator was specified with reference to the literature [27, 31], in which a suggestion is made that the magnitude of the peak-to-peak magnitude of the torque demand of a system is proportional to the amplitude of the harmonic content in its output motion.

Dimension synthesis was performed using a wide range of W_2 values, ranging from 1 to 1000. W_1 was kept constant at 1. The resultant mechanisms were analysed manually. Of the resultant mechanisms, the four mechanisms which best satisfied the kinematic quality indicator, were identified.

Using the Equations (4.4) and (4.5), mass and inertia values for the links of the synthesised mechanisms were estimated. DYSIM models of the four alternative mechanisms

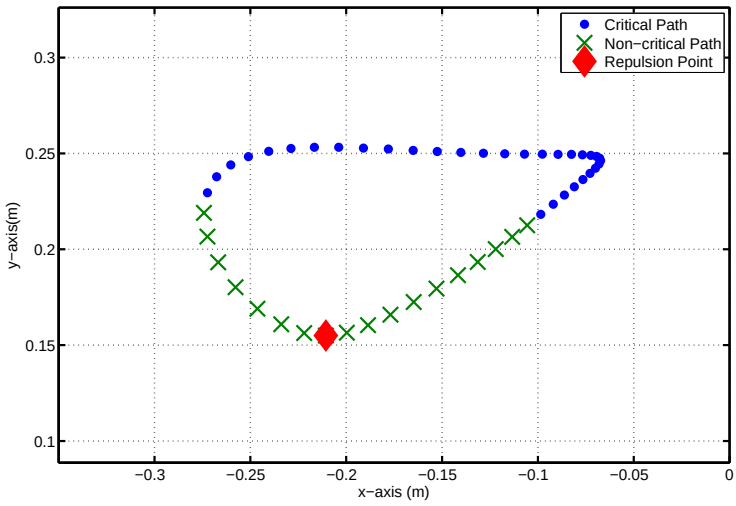


Figure 4.5: The discretised output path of the seed mechanism with the different portions of the path shown

were created. The mechanism was modelled as 7 objects, five links and one triangular tertiary link. When synthesising the dimensions of the triangular tertiary link, it was considered as three separate links, but amalgamated to become a single tertiary link in the DYSIM model. The model of the seed mechanism, as created in Ch. 2 was also considered.

The model of the seed mechanism was created using accurate data taken from a CAD model of the mechanism. This data did not always position the centres of mass of the links at the mid-point of the link. However, when modelling the four alternative candidate mechanisms, the links were considered to have their centres of gravity at their centre point.

In this case of inverse dynamic analysis, the mechanism crank angle was selected as the motion defining variable. It was desired for the mechanism to be simulated operating at a constant cyclic speeds of 100, 200, 450 and 600 rpm.

Using DYSIM, the mechanisms were analysed using forwards and inverse dynamics. Using forwards dynamics, the output orbits of the mechanisms were simulated. These paths were compared to those generated using the traditional synthesis method for each

mechanism, in order to validate each model. Using inverse dynamics the variation in torque that needs to be exerted on the mechanism crank to propel it a constant speed over a complete cycle was estimated.

4.3 Results & Discussion

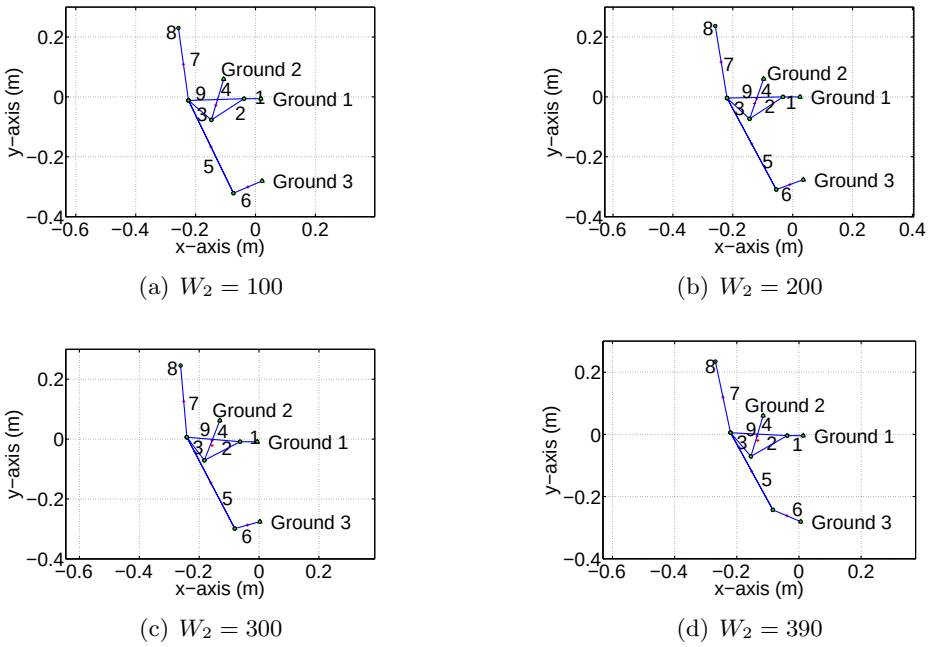
4.3.1 Mechanism Synthesis & Kinematic Analysis

As discussed, the *dimension synthesis* was performed using a variety of values of W_2, whilst W_1 was kept constant. The larger the value of W_2, the stronger the *repulsion* of the repulsion point, resulting in mechanism designs with output paths which should pass ever further away from the repulsion point being requested. This behaviour was borne out in the synthesis results.

As W_2 was increased from 1 up to a value 400, mechanisms were generated with output paths with ever decreasing vertical displacements. The paths of the recoil portions of their motions passed ever further from the repulsion point but still followed the desired output path closely in the critical portion of their motions. As W_2 was increased above 400, the influence of the repulsion point became so significant that mechanisms were generated with output paths that did not correlate well to the form of critical portion of the output path. These mechanisms were deemed unsuitable and discarded.

Furthermore, if W_2 was increased above 390, mechanisms with non-realistic behaviour were generated. Such behaviour includes mechanism links passing instantaneously through inflection points and the breaking of links to allow the mechanism to perform a continuous cycle. A feature of the synthesis algorithm used by SWORDS is to allow mechanism construction constraints to be violated in this way to achieve continuous cyclic output motion. Doing so however, incurs a severe cost penalty in its optimisation routine. Clearly, only feasible mechanisms were considered for dynamic analysis. A number of mechanism designs were analysed for kinematic quality and four designs

were selected with desirable kinematic behaviour and also representing a spectrum of W_2 values. The selected mechanisms are shown in Fig. 4.6.



(a) $W_2 = 100$ (b) $W_2 = 200$

(c) $W_2 = 300$ (d) $W_2 = 390$

Figure 4.6: The selected synthesised mechanism designs

The dimensional parameters of the seed mechanism and the newly synthesised alternative mechanism designs are summarised in Tab. 4.1.

The scaling factors used to estimate the masses and inertias of the synthesised mechanisms are summarised in Tab. 4.2. These factors were derived using data taken from the seed mechanism (Woodpecker mechanism), as detailed in Tab. 2.1 and using Equations (4.4) and (4.5). To derive mass and inertia scaling factors for the two tertiary links, where single mass and inertia values had been predetermined using CAD data, scaling factors were derived by considering these links to be single, uniform, continuous links, the length of which is the sum of the lengths of its constituent parts.

The estimated physical data used to create the DYSIM models of the synthesised mechanisms is summarised in Tab. 4.3.

Parameter	Seed Mechanism	$W_2 = 100$	$W_2 = 200$	$W_2 = 300$	$W_2 = 390$
Length Link 1 (mm)	62.00	56.31	57	57.22	51.53
Length Link 2 (mm)	127.03	130.12	133.62	134.44	134.31
Length Link 3 (mm)	103.00	100.35	102.43	96.85	100.42
Length Link 9 (mm)	188.30	185.66	186.85	178.48	182.58
Length Link 4 (mm)	144.00	142.35	141.77	142.64	136.57
Length Link 5 (mm)	348.90	344.08	346.85	344.58	283.22
Length Link 7 (mm)	245.19	244.25	243.77	240.15	233.97
Length Link 6 (mm)	102.00	104.03	96.12	87.11	97.13
Link 5 to Link 7 offset angle (degs)	14.44	17.98	20.13	22.68	17.95
Ground 1 Normalised Coordinates (mm)	(0, 0)	(0, 0)	(0, 0)	(0, 0)	(0, 0)
Ground 2 Normalised Coordinates (mm)	(-121, 60)	(-124.24, 63.33)	(-131.25, 63.68)	(-86, 70.97)	(-128.95, 64.23)
Ground 3 Normalised Coordinates (mm)	(17.5, -274)	(3.99, -274.67)	(0.27, -273.60)	(8.79, -266.94)	(-8.05, -276.58)

Table 4.1: The dimensional parameters of the seed mechanism design and the four selected synthesised mechanism designs. Note that Links 2, 3 and 9 form a tertiary link. Links 5 and 7 also form a tertiary link. The two links are offset at a fixed angle to one another. For ease of comparison, ground location coordinates have been normalised, such that Ground 1 lies at the origin in each case.

Link Number	Mass Constant (m_{0n})	Inertia Constant (J_{0n})
1	1.50×10^{-2}	2.53×10^{-7}
2, 3, 9	7.41×10^{-4}	2.35×10^{-7}
4	2.88×10^{-3}	1.53×10^{-7}
5, 7	1.10×10^{-3}	1.92×10^{-6}
6	1.26×10^{-3}	2.15×10^{-7}

Table 4.2: The mass and inertia scaling factors

4.3.2 Dynamic Analysis

Using DYSIM, the output paths of the seed mechanism and the four newly synthesised mechanisms were estimated. The output paths of the synthesised mechanisms were compared and validated with results from the kinematic synthesis program. They were seen to correlate well.

A summary of the output paths of the four synthesised mechanisms are compared with the output path of the seed mechanism in Fig. 4.7.

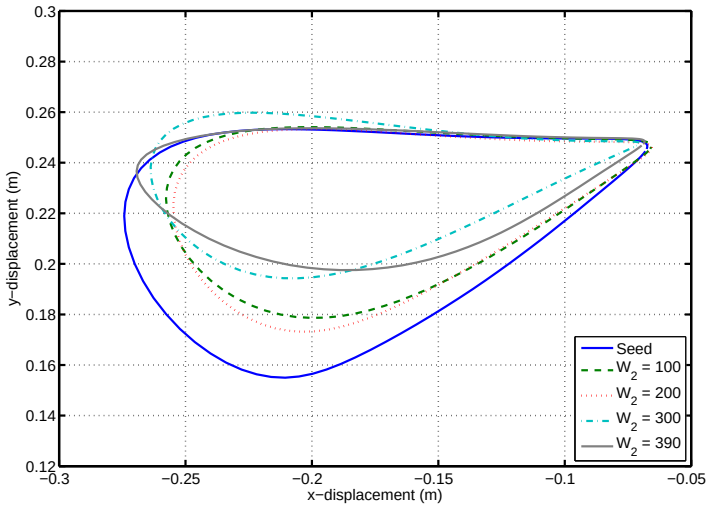


Figure 4.7: The output paths of the synthesised output paths, compared to the desired output path

Using inverse dynamics, the torque demands of each mechanism to cycle each mechanism at a variety of speeds were simulated. It was observed that for a given mechanism design, as cyclic speed was increased, the peak-to-peak magnitudes of the torque demands increased.

Link Number	$W_2 = 100$		$W_2 = 200$		$W_2 = 300$		$W_2 = 390$	
	m (kg)	J (kgm^{-2})	m (kg)	J (kgm^{-2})	m (kg)	J (kgm^{-2})	m (kg)	J (kgm^{-2})
1	0.84	6.75×10^{-4}	0.85	7.00×10^{-4}	0.856	7.08×10^{-4}	0.77	5.17×10^{-4}
2	0.10	3.84×10^{-4}	0.10	4.16×10^{-4}	0.10	4.23×10^{-4}	0.10	4.22×10^{-4}
3	0.74	1.76×10^{-4}	0.08	1.87×10^{-4}	0.07	1.58×10^{-4}	0.07	1.76×10^{-4}
4	0.41	1.27×10^{-3}	0.41	1.25×10^{-3}	0.41	1.27×10^{-3}	0.39	1.12×10^{-3}
5	0.38	8.65×10^{-2}	0.38	8.86×10^{-2}	0.38	8.69×10^{-2}	0.31	4.82×10^{-2}
6	0.13	3.07×10^{-4}	0.12	2.42×10^{-4}	0.11	1.8×10^{-4}	0.12	2.50×10^{-4}
7	0.27	3.09×10^{-2}	0.27	3.08×10^{-2}	0.27	2.94×10^{-2}	0.26	2.72×10^{-2}
9	0.13	1.11×10^{-3}	0.13	1.14×10^{-3}	0.13	9.90×10^{-4}	0.14	1.06×10^{-3}

Table 4.3: The scaled mass and inertia values for the synthesised mechanism designs

Figure 4.8 shows a summary of the variations in torque, which a drive motor would need to exert on the mechanism crank of each mechanism to drive it at a constant speed of 600 rpm.

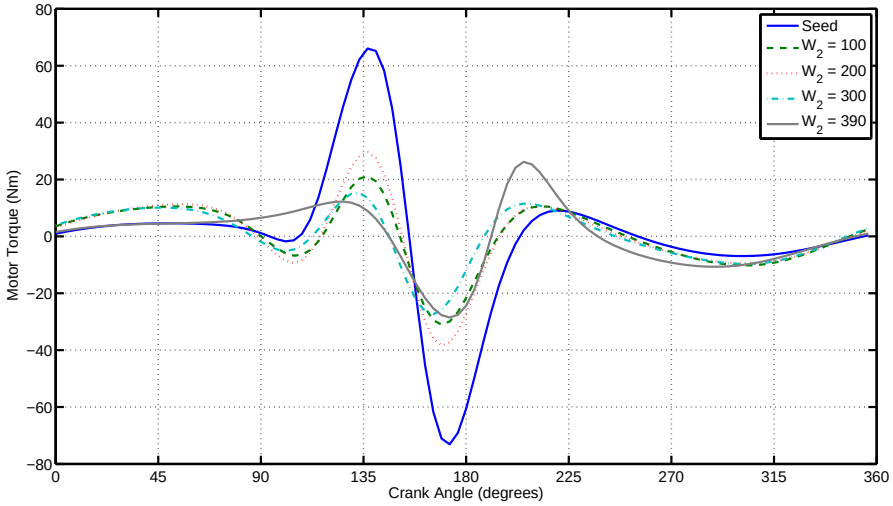


Figure 4.8: The variation in driving torque needed to propel the cranks of the seed mechanism and synthesised mechanism at 600 rpm over a complete cycle of crank

It is noticeable from Fig. 4.8 that the magnitudes of the peak-to-peak torque demands of all of the synthesised mechanisms are far smaller than that of original seed mechanism. The mechanism generated using $W_2 = 300$ exhibited the cyclic torque demand with the smallest peak-to-peak magnitude and is therefore the most desirable. Compared with the equivalent demand of the seed mechanism, this mechanism exhibited a demand with a peak-to-peak magnitude reduction of 70%. The next most effective mechanism was generated using $W_2 = 100$ (63%), followed by $W_2 = 390$ (61%) and finally $W_2 = 200$ (51%). The form of the torque demand signals and hierarchy of effectiveness were replicated at lower actuation speeds, as shown in Fig. 4.9.

Work was carried out to analyse and compare the harmonic content present in the output motion of the most superior alternative mechanism, $W_2 = 300$, to the harmonic content present in the output motion of the seed mechanism. In an identical manner to that carried out in Sec. 3.2.1, the angular motion of the end effector bearing link was compared with respect to the motion of the mechanism crank. The cranks of both mechanisms were simulated being actuated at a cyclic rate of 600rpm. Analysis was

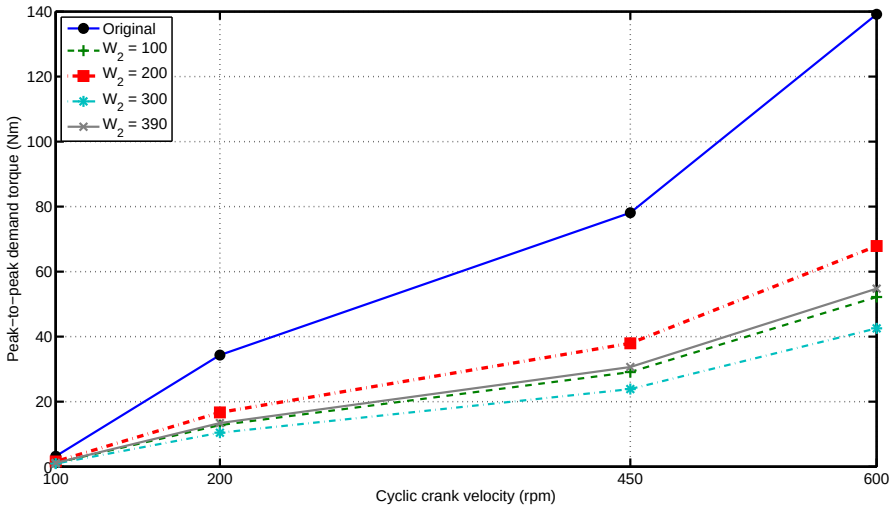


Figure 4.9: A comparison of demand torque peak-to-peak magnitude values needed to propel the cranks of the seed mechanism and synthesised mechanism at a variety of velocities over a complete cycle of crank

performed using Fourier analysis. The resultant comparison is shown in Fig. 4.10

Analysis of Fig. 4.10 shows that the amplitude of the harmonic content present in the output motion of the most superior alternative mechanism is consistently smaller than the amplitude of the equivalent motion of the seed mechanism. This difference is most significant at high frequencies. The result satisfies the aim of the investigation.

At the heart of the method discussed in this chapter, is the ability to modify the desired output path in order to achieve new mechanism designs. The scope to be able to do this is dictated by the length of the defined non-critical output path. The longer the non-critical portion of the output path, the greater the variety of different output paths that can be explored and the greater the variety of mechanism designs that can be analysed. This increases the probability that a superior mechanism design can be identified. However, with the presence of more alternative candidate mechanisms, comes increase an in processing time.

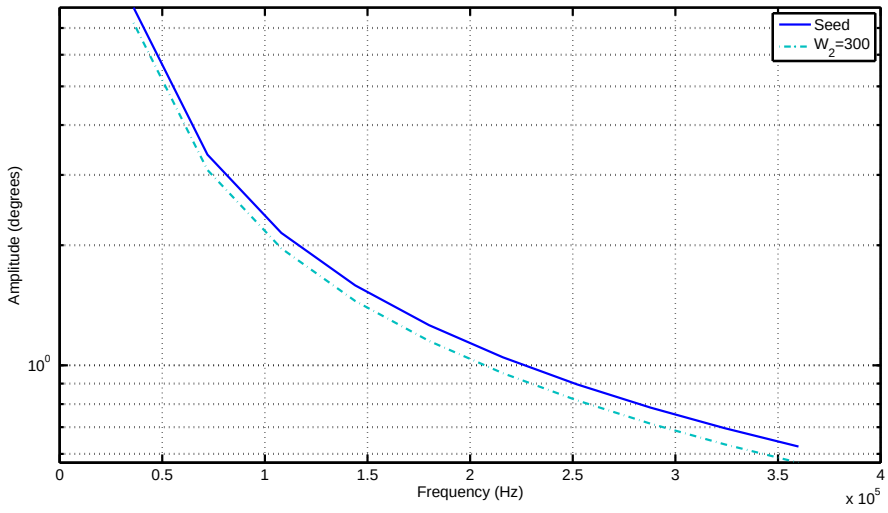


Figure 4.10: A comparison of the harmonic content present in the output motion of the seed mechanism and the most superior alternative mechanism $W_2 = 300$ in response to a 600 rpm constant velocity demand signal.

4.4 Conclusions

The work in this chapter proposes a unique and highly effective strategy to combine a traditional mechanism synthesis process with a dynamic analysis process. This integrated approach can be applied to generate mechanisms capable of following complex output paths with good dynamic characteristic when operating at high speeds.

The strategy can be broken down into five key stages:

1. **Problem Definition**: In this stage of the method, *number synthesis* is carried out to specify the style of the mechanism to be considered. The desired output path is also specified in this stage of the method. The output path should contain regions of non-critical motion. In this region, it should be possible to vary the form of the output path without adversely affecting the functionality of the mechanism. Kinematic quality indicators must also be specified.

2. **Mechanism Synthesis**: In this stage of the method, *dimension synthesis* is carried out to determine the dimensions of the mechanism links. A point near

to the non-critical portion of the desired output path was selected to act as a repulsion point. A weighting factor, W_2, varies the influence of the repulsion point on the synthesis process. As the weighting factor is increased, the repulsion point becomes more influential and the synthesis process seeks to generate mechanism designs with output paths that pass ever further away from the repulsion point. By iteratively varying W_2, a wide variety of alternative mechanism designs can be obtained.

3. **Kinematic Analysis**: Of the synthesised mechanisms, those with the best kinematic performance are manually selected in accordance with pre-determined kinematic performance criteria.

4. **Modelling and Parameter Estimation**: In this stage of the process, mass and inertia scaling factors for each link of the synthesised mechanisms are derived. These values can be derived using data from CAD models or using an alternative method. Using these factors, mass and inertia values for each link of the newly synthesised mechanisms can be estimated. This data is used to create dynamic models of the synthesised mechanisms.

5. **Dynamic Analysis**: Using the dynamic models, forwards and inverse dynamics are used to analyse the mechanisms. Dynamic performance indicators were used to identify the most superior alternative design.

The proposed approach was successfully applied to the design of the Woodpecker mechanism. The configuration of the Woodpecker mechanism was adopted as the seed mechanism and its output path adopted as the desired output path. A non-critical portion of the output path was identified in the desired output path. Using the *dimension synthesis* process described in this chapter, four alternative mechanism designs were selected, each with different output paths, but which all satisfied the kinematic quality indicators. Dynamic analysis of these mechanisms identified a single mechanism design that demonstrated outstanding dynamic qualities compared to the other mechanism designs. The peak-to-peak magnitude of the torque demand signal needed to actuate the mechanism crank of this new mechanism at a constant cyclic rate of 600 rpm was

simulated to be 70% smaller than the equivalent demand of the seed mechanism. The harmonic content of the output motion of this alternative mechanism was compared to the amount of harmonic content in the equivalent motion of the seed mechanism. The comparison indicated that the most superior alternative mechanism had an output motion with smaller amplitude harmonic content than the seed mechanism reiterating the potential effectiveness of the method. These results demonstrate the effectiveness of the method. The relationship between peak-to-peak torque magnitude and harmonic content amplitude is also re-confirmed.

4.5 Further Work

The work described in this chapter demonstrates the potential benefits of adopting a process in which mechanisms are synthesised for both kinematic and dynamic quality. A process of synthesising such mechanisms was proposed to illustrate this point.

When performing this work, the synthesis process was controlled by varying the parameter value W_2, the values of which were arbitrarily selected by the user during the investigation. Using these values alternative mechanisms designs were generated which were superior to the seed mechanism. One mechanism design was identified as being the most superior. It is possible that a even better mechanism design may exist, generated using a value of W_2 not tested. The most efficient way of identifying this value of W_2 is to consider the complete strategy as a cost-based optimisation problem using W_2 as the independent variable. With further work, the strategy described in this chapter could be reformulated to take the form of an optimisation problem. A flow diagram of a proposed method is shown in Fig. 4.11.

The method depicted in Fig. 4.11 is clearly closely based on the method process discussed earlier in this chapter, with each mechanism design being analysed first for kinematic quality and then for dynamic quality as before. It can be seen that method works by performing iterations using different values of W_2. For each value of W_2

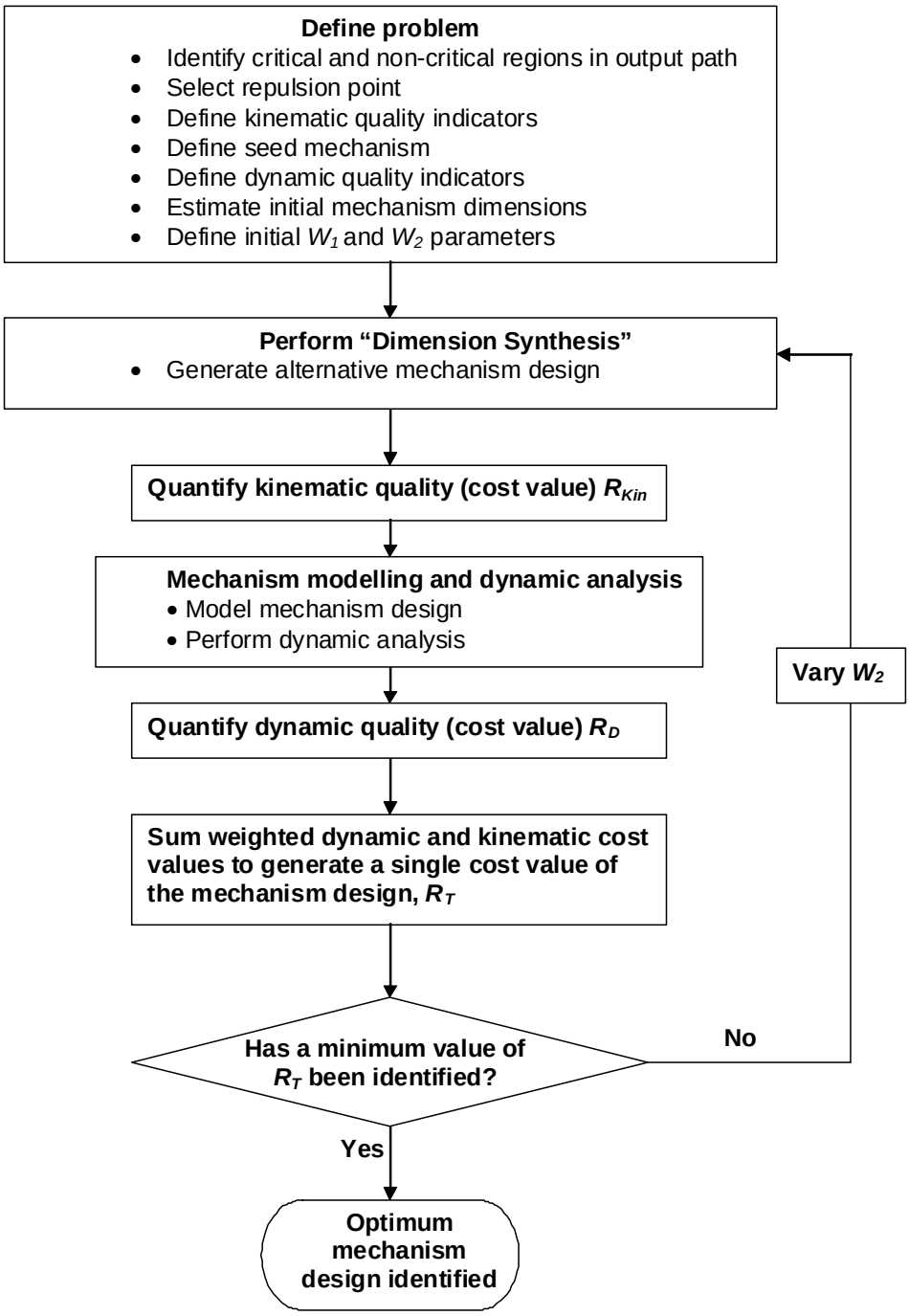


Figure 4.11: A proposed method of integrating the combined kinematic and dynamic mechanism synthesis method into a numerical parametric optimisation routine

the resultant mechanism design is analysed for both dynamic and kinematic quality. Numerical dynamic and kinematic weighted quality indicating cost values, R_D and R_{Kin} respectively, are generated. These values and then and then summed to form a single unified cost value, R_T. In all cases, the lower the cost value, the better. The optimisation process seeks to minimise R_T.

Critical to the functionality of this optimisation process is the definition of the kinematic and dynamic quality indicators in the form of clear numerical functions. Doing this totally automates the process, negating the need for manual intervention to perform the procedure discussed in this chapter. The development of such an optimisation routine would form a logical progression of the work discussed in this chapter and would result in the creation of potentially powerful mechanism synthesis tool.

Chapter 5

Design and Modelling of The Reconfigurable Test Rig

The work described in this chapter builds on the work discussed in Ch. 4, in which new mechanism designs were synthesised with varying degrees of dynamic quality. In this chapter, work is described detailing the development and construction of a laboratory test rig with which it is possible to experimentally verify the work detailed in Ch. 4. The test rig was designed to be reconfigurable in order to facilitate the assembly of a wide variety of different mechanism designs, with differing configurations and link lengths, without the need to refabricate mechanism links and other constituent parts. Work was also carried out to model this new reconfigurable test rig.

5.1 Mechanism Construction

The primary purpose of this test rig was to provide a method of constructing a wide variety of mechanism designs without the need for extensive refabrication of parts. By actuating the mechanism, the dynamic and kinematic behaviour of the mechanism design could be analysed. To make this test rig fit for purpose, it had to possess a

number of specific features. Most critically, the test rig had to be versatile and possess the ability to be easily assembled in different configurations. It must also possess the ability to simulate links of differing length and differing ground point locations without the refabrication of parts. One of the ground points must be connected to a motor to enable the mechanism to be propelled. The test rig must also facilitate the measurement and logging system behavioural data (motor velocity, motor torque etc.) to allow the response of the system to a demand signal to be analysed.

With this in mind, a test rig was designed which satisfied all of the aforementioned design goals. A photograph of the complete test rig is shown in Fig.5.1.

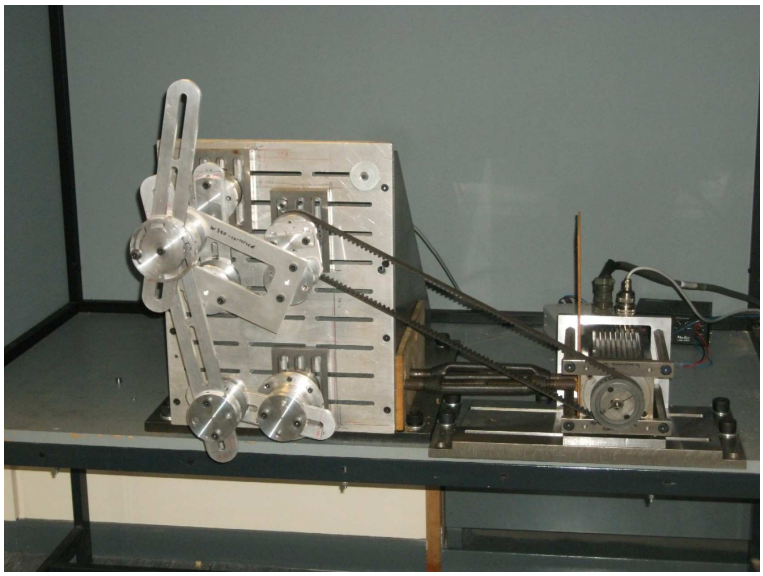


Figure 5.1: The reconfigurable test rig

In summary, the test rig consists of a series of slotted aluminium links, which are connected together using a two-part rolling connection assembly. In this way, adjacent links can be bound to each other but still be able to rotate relative to one another. A detailed sectioned view of a rolling connection unit can be seen in Fig. 5.2.

The main body of a rolling connection consists of two aluminium components, an axle component and a housing sleeve component which sits around the axle. Two SKF 161101 deep section ball bearing units [69] are mounted on the axle shaft to permit free, rotary motion between the housing and the axle components. Spacers are placed

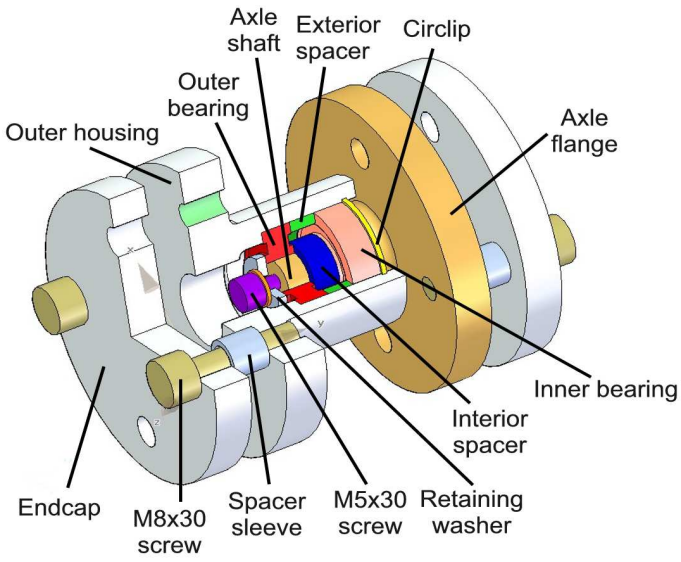


Figure 5.2: A sectioned view of a rolling connection assembly

inside the housing unit between the bearing units to locate them and to prevent the races of the inner and outer bearing units fouling on one another. Shoulders on the housing locate the outer race of the outer bearing. It is located on the axle shaft by a M5x30 mm screw and a washer, which lies flush against its inner race. The outer race of the inner bearing is located using an internal circlip. It inner race is located by a shoulder at the base of the axle shaft.

To attach a link to either the axle or the housing component, two M8x30 mm screws are passed through 8 mm wide plain holes in an endcap component. These screws then pass through a slot in the link and into tapped M8 holes in the large outer flanges on either the housing or the axle unit. Sufficient torque is applied on the screws to hold the link firmly in place. Spacing sleeves are positioned around the screws, between the endcap and housing or axle unit flange to locate the screw in the link slot. Using the slots in the link, the rolling link can be positioned anywhere along its length allowing the effective length of the link to be varied. An example of this assembly method is shown in Fig. 5.3.

To support the mechanism, a 20 mm thick aluminium baseplate was used, which is mounted onto a supporting structure made from 20 mm thick steel. Such a substantial

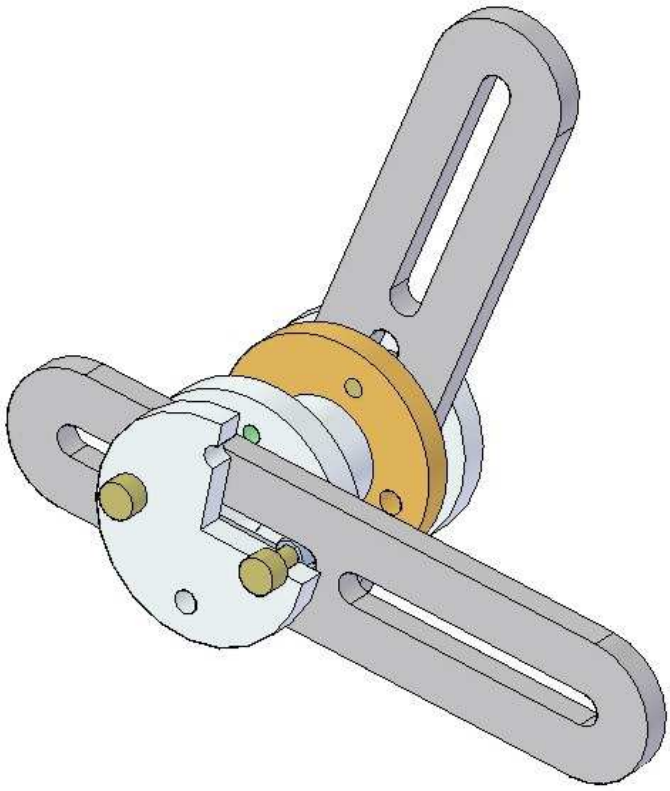


Figure 5.3: A link attached to a rolling connection assembly

and rigid mounting construction was necessary to minimise deflection in the structure under what could be considerable dynamic and static loading from the mechanism. The supporting structure was fixed to a strong, sturdy and rigid table, using a number of M20 bolts. A CAD drawing of the baseplate and supporting structure is shown in Fig. 5.4.

As shown in Fig. 5.4, a series of 8.5 mm wide slots were cut in the aluminium plate. The slots enabled the mechanism to be attached to the baseplate via rectangular steel grounding plates, with each grounding plate allowing for a single ground point for the mechanism. In each plate, three 8.5 mm wide slots were cut. These were then counterbored to a width of 12 mm and to a depth of 10 mm. Four M5x20 mm holes were also drilled into the top surface at the plate. For each grounding plate, three steel nuts were manufactured from a length of 12 mm wide hex bar. An M8 thread was tapped into each nut. The vertical height of each nut was 8 mm, and therefore less than the depth of the counterbore on the grounding plate.

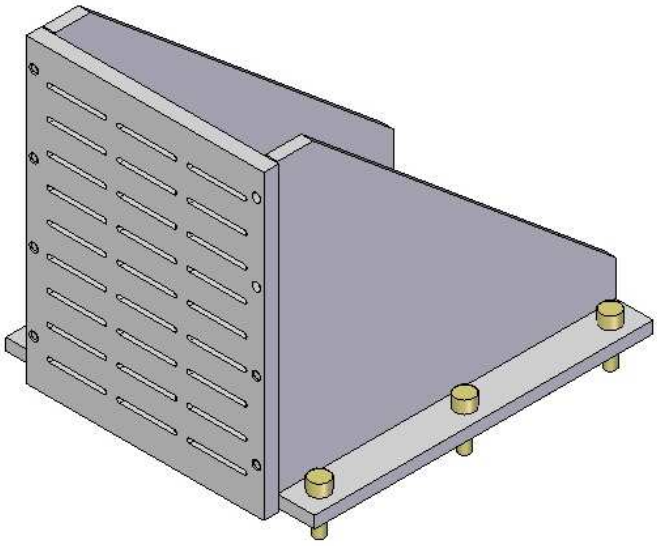


Figure 5.4: The baseplate and supporting structure

M8 screws were inserted through the rear face of the baseplate and through the slots in the grounding plate and held in place by the steel nuts. Since the width of the nuts corresponds to the width of the counterbores, when a torque is applied to the nut, its rotation is restrained by the counterbore, allowing for the screw to be wound tightly into the nut using only a single Allen key. It was intended that the slots of the grounding plates should be positioned perpendicular to the slots in the baseplate. The slots cut into both components were spaced in such a manner that no matter where on the front face of baseplate a grounding plate was positioned, it was always possible to pass a screw through the baseplate and through at least two of the slots of the grounding plate, allowing for a secure connection between the two components. The third slot may be fully or partially occluded by the baseplate. In this way a grounding plate is infinitely locatable on this baseplate. An example of this method of assembly is shown in Fig. 5.5.

To attach a mechanism link to a grounding plate, a rolling link assembly, very similar to that used to connect two links was used. The assembly differs only in that the spacing of the holes on the flange of the axle component is different to standard axle components. The holes are also smaller in diameter, as they are designed to act as

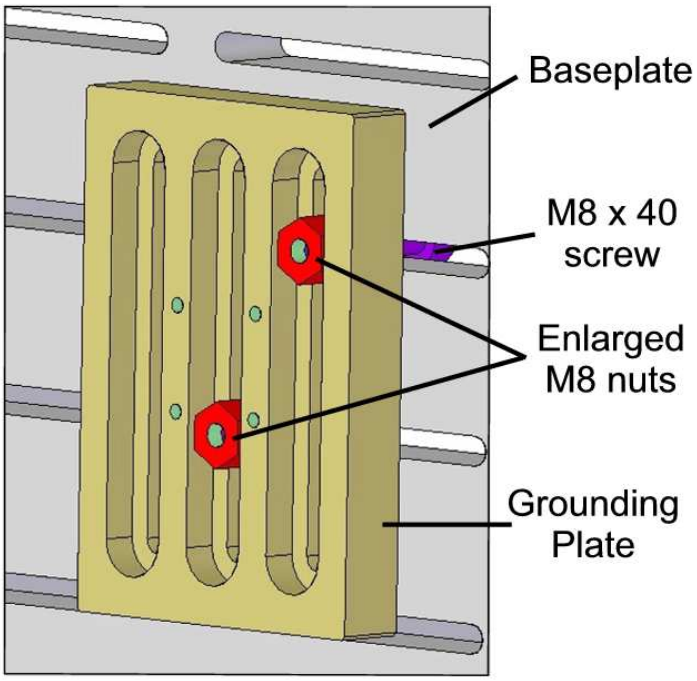


Figure 5.5: Attaching a grounding plate to the baseplate

through holes for M5 screws. They are counterbored to a depth of 5 mm to allow for the complete submersion of the head of an M5 screw. These holes align with four M5 tapped holes cut into the grounding plate, as shown in Fig. 5.5. The axle component is screwed onto the grounding plate using M5x30m screws.

When assembling the mechanism, it may be necessary to have a grounding link with a horizontal length twice or even three times the length of single rolling link. To allow for this eventuality, an offset assembly was designed. In Fig. 5.6 the assembly used to offset a link by the length of a single rolling link is depicted. In this example, the offset assembly is connected to a ground plate.

The offset assembly consists of two components; a male component and a female component. The flat outer face of the female piece mates to the flat face on the housing component and is held in place by four M8x20 mm screws, which interface with the M8 tapped holes on the flange of the housing and via four counterbored 8.5 mm diameter holes in the female offset component. The counterbore is of sufficient depth to allow the screw heads to be fully submerged in the female offset component. Drilled in the

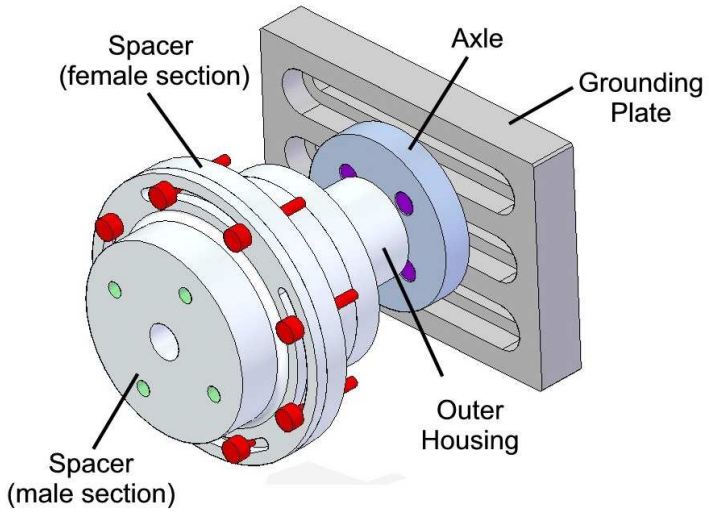


Figure 5.6: A double length connection connected to a grounding point

flange of the female offset piece are eight M5 tapped through holes, equispaced around the circumference. The male offset piece mates to the surface female offset piece, as depicted. Four 5.5 mm wide through slots were cut in the flange on the male piece. Eight M8x20 mm screws were passed threw the flange and into the tapped holes in the female component, as shown. A mechanism link can be connected to the outer surface of the male offset component in the same manner as has been done previously.

If a grounding link of additional lengths are required, a supplementary spacing components can be included in the assembly depicted in Fig. 5.6, positioned between the male and female spacer components. Such a component is depicted in Fig. 5.7.

It was identified that to assemble the Woodpecker mechanism, it was not necessary to use supplementary spacing components.

With reference to the schematic of the Woodpecker mechanism, as shown in Fig. 2.2, it can be seen that one of the features the Woodpecker mechanism is the positioning of the end effector (Link 8) at the end of a Link 7. Link 7 lies at a fixed angle to Link 5. To allow two links to be assembled at a fixed angle, the assembly depicted in Fig. 5.8 was used.

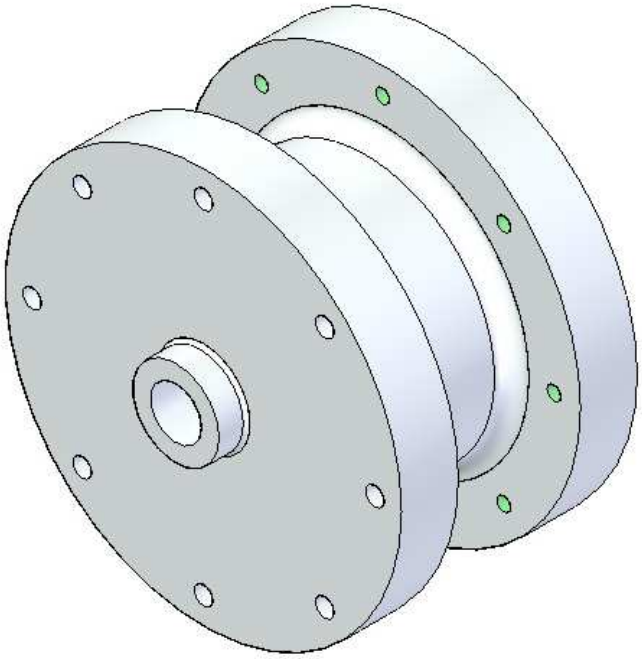


Figure 5.7: A single length supplementary spacing component

This assembly functions as follows. Typically, as shown in Fig. 5.3, a circular endcap is used to clamp a link to a rolling joint. To support a fixed angle connection, this endcap is replaced with an alternative part, the fixed angle inner part. Fundamentally, the fixed angle inner part is an enlarged endcap, both in terms of width and thickness. Four M8 tapped through holes were drilled in the component, two of which were counterbored to a sufficient depth to allow submersion of an M8 screw head. To attach the fixed angle inner part to the outer housing of the connection joint, M8x30 mm screws can be passed through the two counterbored holes and into the tapped holes in the flange of the outer housing component, securing a link to the rolling joint in place of a standard endcap.

With the counterbores cut to a sufficient depth to allow these screws to be fully submerged, a flat surface on the exposed outer face of the fixed angle inner part is presented onto which the second link can be mated and positioned at the desired angle. The outer section of the fixed angle joint was then mated to the outer surface of the second link. Three M8x30 mm screws were passed through the fixed angle outer part and into the

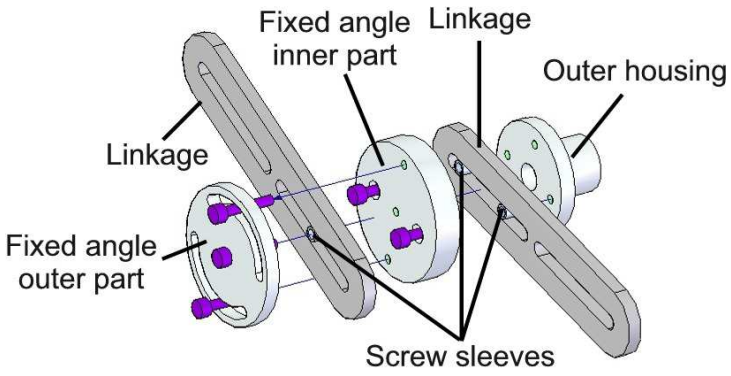


Figure 5.8: An exploded view of the fixed angle connection assembly

fixed angle inner part where they interface with three M8 tapped holes on the fixed angle inner part. Two screws pass through the slots on the outside of the outer part, whilst one is inserted through the central hole and through the slot in the link. A sleeve is placed around this screw to take up slack contact with the link slot.

5.1.1 Mechanism Propulsion And Control

The design process of the reconfigurable test rig was performed after the development and experimental work on the Woodpecker mechanism test rig, detailed in Ch. 3, had been completed. Through the development of the Woodpecker mechanism test rig, an effective motion control and mechanism propulsion strategy was identified. It was identified that it would be feasible to adapt this strategy for use with the new reconfigurable test rig. The main parts of the control and propulsion strategy, namely the Electro-Craft S-4075 servo motor, the BRU-DDM30 motor drive and Deva004 controller, were removed from the Woodpecker mechanism test rig and installed on the new reconfigurable test rig. The software that was written to control the Woodpecker mechanism test rig was also deemed suitable for reuse with the reconfigurable test rig.

As before, a belt drive transmission system was used to transmit power from the connect the drive shaft of the servo motor to the crank shaft. Again a synchronous, toothed drive belt was used. Suitable pulleys were attached to the output shaft of the servo-

motor and to the outer housing of the rolling link attached to the grounding plate to which the crank link is attached. The two pulleys were connected using a suitable belt. A gear ratio of 1:1 existed between the two pulleys.

Peculiar to this test rig is the necessity for the motor to be firmly grounded but also to be relocatable. To achieve this, the assembly detailed in Fig. 5.9 was used.

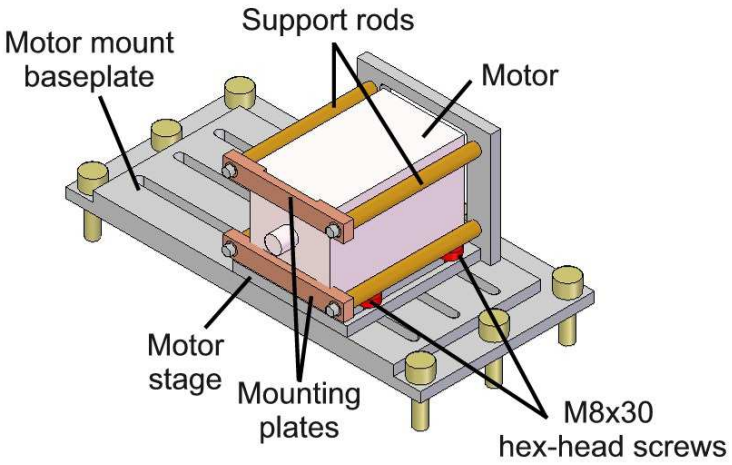


Figure 5.9: Mounting the motor

This setup consists of two main sections, a rigid steel base attached to the table surface and a smaller assembly on which the motor sits. The smaller assembly sits on the surface of the base. The steel base has two 12 mm slots cut in it, which span its width. These slots are counterbored to a width of 15 mm on the underside of the base to a depth of 12 mm. Four 15 mm wide square components were manufactured with M10 threads cut in them. These components act as retaining nuts. Two of these nuts were positioned in each slot prior to fixing the base to the table surface. The steel base was fixed to the table surface by passing M20 bolts through 22.5 mm diameter holes in the flange on the steel base component and corresponding holes drilled in the table surface. They are secured on the underside of the table with suitable nuts.

The smaller upper section is closely based on the motor mounting method used previously on the Woodpecker mechanism test rig, with some parts being reused. The horizontal base on which the motor sits was positioned on the top surface of the large

steel base attached to the table surface. The smaller base has two 15 mm wide slots cut in it. When positioned correctly, the slots in the smaller base and the slots in the larger base lie perpendicular to one another. Four M10 hex-head screws, two in each respective slot, were then passed through the slots in the smaller base, through the larger base and interfaced with the large square nuts in the counterbore under the larger base. In this way, it was possible to tension the screw against the nut using a single spanner, rotation of the nut being restrained by the counterbore of the large steel base. The slots in the larger base allowed the motor to be positioned anywhere along the length of the base in a direction parallel with the aluminium baseplate onto which the mechanism is attached, whereas the slots in the upper base section allow the position of the motor be adjusted in a direction perpendicular to the baseplate.

To take up the majority of the slack in the drive belt, the motor was firmly pushed in a direction away from the baseplate, fixing it in place. To provide additional belt tension, a pair of dual sided extension rods were positioned between the motor and the baseplate as shown in Fig. 5.10. These components were extended as required, to generate sufficient tension.

Figure 5.10: The belt tensioning extension rods

5.1.2 Data Collection

The data collection strategy developed through the design of the Woodpecker mechanism test rig was adapted for use with the reconfigurable test rig. The data acquisition card (DAQ) and data logging laptop computer, used previously to collect data from the BRU-DDM30, was reused. As before, actual motor position P_{out} information, as well as motor output torque data, T_{out} (logged as motor command current) was collected by the DAQ and logged by the laptop. An accelerometer was also installed on the test rig, as shown in Fig. 5.11 allowing the amplitude of vertical vibrations experienced by the baseplate, G_{out}, to be measured. Data from the accelerometer was also logged using the DAQ.

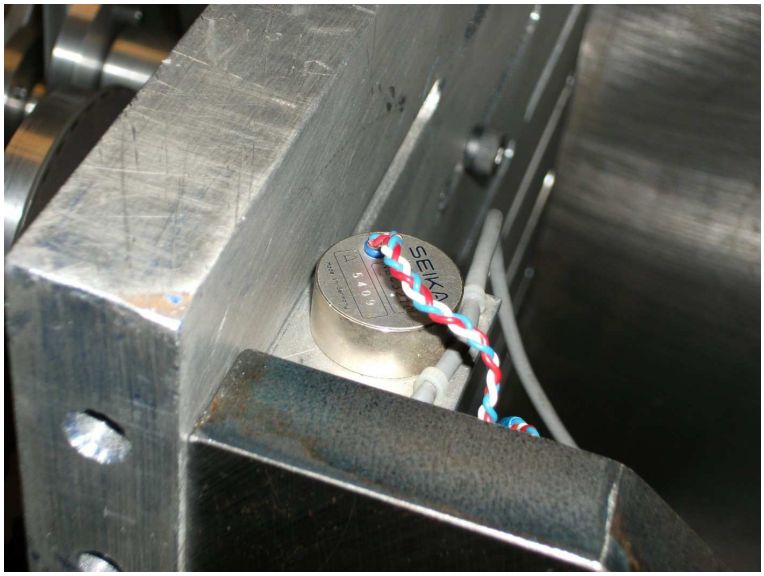


Figure 5.11: The accelerometer

Data was output from the accelerometer in the form of a voltage signal, with a 1 Volt output corresponding to a 1g acceleration.

Demand and actual velocity output signals, V_{dem} and V_{out} respectively and demand and actual position output signals, P_{dem} and P_{out} respectively were all logged using the control compute via the Deva004 control card, as before. Other than these minor modifications, the data logging strategy adopted by the reconfigurable is identical to that used by the Woodpecker mechanism test rig, as described in Sec. 2.3.1.

5.1.3 Guards

To isolate the test rig from the user and to provide protection should the mechanism fail and parts of the mechanism become detached from the test rig during operation, custom designed safety guards were manufactured. The guards consisted of two assemblies one of which was bolted to the table surface and the other was freestanding and supported by castor wheels positioned at the four corners of the bottom of the supporting structure of the guard. For both sections, supporting frames were constructed from steel, *Speedframe* tubing [70]. Panelling was then attached to the outer sides of the frame. The panelling of the portion of the guard attached to the table top consisted entirely of MDF. For the freestanding, mobile portion of the guard, 10 mm thick polycarbonate sheets were used.

Figure 5.12: The guards

The mobile portion of the guard was designed in such a way that the centre of gravity of the guard lies directly through the centre of the construction, equidistant between all four castor wheels. This feature makes the structure particularly stable when being moved. *Suitcase latch* style fasteners were used to join the two portions of the guard together during operation.

The BRU-DDM30, breakout board and DAQ were attached to the side of the MDF portion of the rear section of the guard. Holes were cut in the MDF to allow cables to be fed through the guard to the equipment inside.

5.2 Assembling the Mechanism

Prior to manufacturing the parts for the test rig, a model of the complete reconfigurable test rig was created using CAD software. With the alternative mechanism designs discussed in Ch. 4 being taken into consideration, CAD models of each mechanism design in the operating environment were created. When positioning the mechanisms on the base plate, care was taken to arrange the grounding plates in such a way that all the mechanism designs could be achieved without the driven ground point having to be moved. This was done to ease later experimental work by negating the necessity for the reassembly of the belt drive transmission when performing tests to compare the alternative mechanism designs. This not only saved time but also allowed the belt tension to be kept constant between each test.

This preparatory work demonstrated that by using the parts in their current form, it was possible to successfully assemble all the mechanism designs on the baseplate with the exception of the design corresponding to $W_2 = 300$. In this case the ground points were located too close together to be realisable using the grounding plates in their current form. With three other alternative mechanism designs achievable however, it was decided that it would be acceptable to merely neglect this mechanism configuration in the experimental work.

With reference to Ch. 4, it can be seen that the an important feature of the mechanism is the triangular tertiary link, to which many of the links of the mechanism connect. The dimensions of this component vary with mechanism design. Due to the complexity of this part, it was deemed that it would be simplest to manufacture a series of discrete one-off parts, with connection holes drilled in the correct places. One part was created

for each mechanism design. A lightening hole was cut in the component. The location and dimensions of this hole were kept constant in each design. An example of such a component is shown in Fig. 5.13.

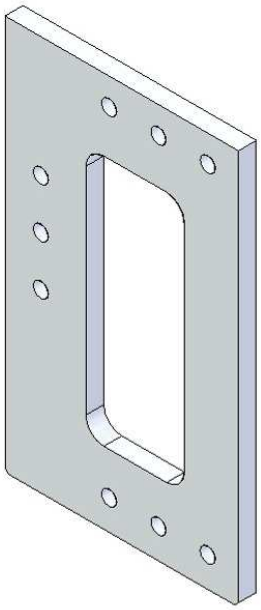


Figure 5.13: A triangular tertiary link component

Having positioned the mechanisms on the baseplate, the motion of each mechanism was simulated using the CAD software. The CAD software analysed the mechanism in motion, identifying part collisions. Parts were modified and even replaced to remove these part collisions and permit free motion. The following parts of were identified for modification or substitution.

1. To prevent part collisions, it was necessary to truncate the width of the circular endcaps used in the rolling joints used in conjunction with Link 4. The endcaps were cut along their hemispheres as shown in Fig. 5.14.

   These endcaps were truncated to 50 mm wide and used in place of normal circular endcaps as needed to prevent part collisions.

2. After analysing the mechanism designs detailed in Ch. 4, it was observed that the cranks of the synthesised mechanism designs were very short in length. It was noted that it was not possible to achieve such short crank lengths using a

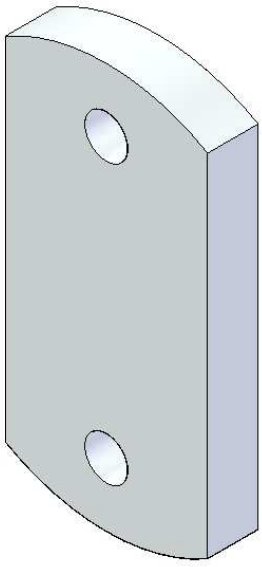


Figure 5.14: A truncated endcap

standard link and rolling joint construction. To remedy this, a new link was designed to permit these crank lengths to realised. This new crank link design is shown in Fig. 5.15.

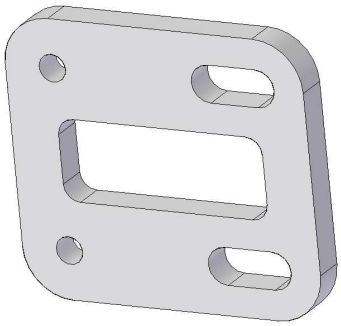


Figure 5.15: The short crank link

To accompany this new connection two other parts needed to be modified:

(a) In the same way as described above, one endcap had to be truncated to a width of 25 mm.

(b) The housing section of a rolling connection was truncated to 50 mm in width as shown in Fig. 5.16. This component was used in place of standard housing component in rolling connection assembly as shown in Fig. 5.2.

The method of assembly of these parts are shown in Fig. 5.17.

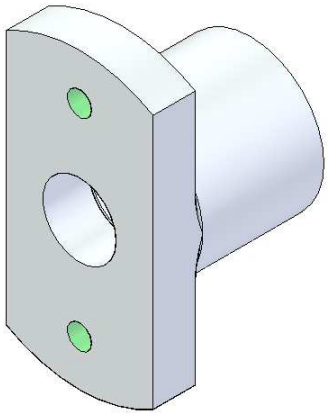


Figure 5.16: The truncated housing component

With reference to Fig. 5.15, it can be seen that cut in the short crank are two horizontal slots and two plain holes. The truncated housing component was attached to the short crank using two M8 screws passed through the short crank and interfacing with the two tapped holes in the truncated housing. Using the 25 mm wide truncated endcap, the short crank was bound to the housing section of the driven ground connection using two M8 screws. Sleeves were positioned around the threads of the screws. Using the slots, it was possible to vary the distance between the 25 mm truncated endcap and the truncated housing, thus varying the effective length of the crank. The truncations were necessary to allow the housing and the truncated endcap to be positioned sufficiently close together to achieve all the desired crank lengths.

Prior to manufacturing these parts, they were tested for suitability using CAD simulations. Once it had been ascertained that the modifications were suitable, the parts were modified. Once all the parts of the test rig had been manufactured, the test rig was assembled. For safety, before any mechanism was actuated using the motor, the mechanism actuated slowly by hand. In this way it was possible to ensure that the mechanism had been correctly assembled and that no part collisions would occur. Once it had been ascertained that the mechanism was correctly assembled, the propulsion of the mechanism was then performed by the motor. The motion control scripts detailed in Ch. 2 were used to control the motion of the test rig, as well as to log system

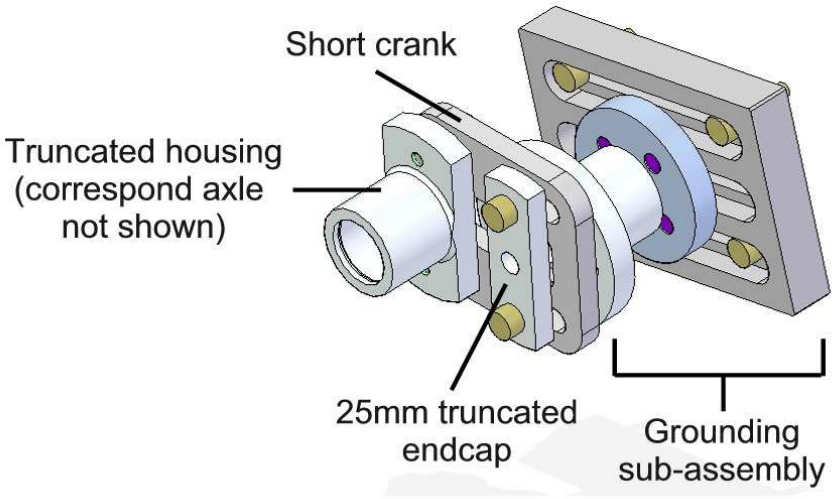


Figure 5.17: Mounting the short crank

behavioural data.

5.3 Modelling the Test Rig

Before any experimental work could be carried out, it was necessary to model the test rig. The process of modelling the reconfigurable test rig was very similar to the method used to model the Woodpecker mechanism rig discussed in Ch. 2. The servo motor, motion controller and drive unit used in this test rig were identical to those used in the test rig described in Ch. 3, and so the models developed for these components were reused.

Models for the new mechanism designs were constructed using DYSIM. Mass, inertia and centre of gravity positions for each component of the system were estimated using the CAD models of the individual parts of the system.

Up to this point in the investigation, when modelling mechanisms, the mass and inertia of the rolling joints connecting the links together had been neglected. This was deemed

acceptable, since the masses and inertias of the rolling joints were negligible compared to the masses and inertias of the links themselves and therefore of little effect on system dynamics. In the case of this new test rig however, the mass and inertia of the rolling bearing connection assemblies are considerable when compared to the masses and inertias of the links and so could not be neglected. For example, in the case of the reconfigurable test rig, a link weighs in the region of 0.27 to 0.50 kg, depending on length, whereas a rolling connection assembly, such as the one depicted in Fig. 5.2, weighs 0.76 kg.

To model a rolling bearing joint assembly, each assembly was broken into two sub-assemblies, one representing the housing portion and the second representing the axle portion. Estimates were made as to which internal components would move in relation with the axle and housing assemblies respectively. Using CAD models it was possible to estimate single, unified mass and inertia values for each subassembly. A variety of mass and inertia values had to be derived due to variation in sub-assembly construction. For example some sub-assemblies incorporate truncated components, whilst others do not and so on, clearly leading to variations in mass and inertia. A summary of the mass and inertia values of the constituent parts of the reconfigurable test rig are summarised in Tab. 5.1.

The assemblies detailed in Tab. 5.1 take into account all associated constituent components such as fasteners, spacers, bearings, and so on.

Using these resultant mass and inertia values, the models of the four mechanism designs were modified to include these subassemblies. These subassemblies were considered as point bodies positioned at the connection points between their relevant links. These bodies were configured to such that the moved and rotated with the connection point as they would in real life. Subassemblies connected directly to ground plates can clearly only perform rotary motion and cannot translate in the plane of motion. For simplicity, these subassemblies were considered as point inertias.

Using this data, inverse dynamic analysis was used to estimate the variations in torque

Part Name	Mass (kg)	Inertia (kgm^{-2})
Short Crank Link	0.20	4.06×10^{-4}
250mm Link	0.26	4.18×10^{-3}
360mm Link	0.38	1.35×10^{-2}
470mm Link	0.50	3.15×10^{-2}
Seed Triangular Tertiary Link	0.62	9.64×10^{-3}
$W_2 = 100$ Triangular Tertiary Link	0.62	9.64×10^{-3}
$W_2 = 200$ Triangular Tertiary Link	0.62	9.64×10^{-3}
$W_2 = 390$ Triangular Tertiary Link	0.62	9.64×10^{-3}
Housing Assembly (with Endcap)	0.41	2.80×10^{-4}
Housing Assembly (no Endcap)	0.28	1.70×10^{-4}
50mm Truncated Housing Assembly (no Endcap)	0.22	1.06×10^{-4}
Axle Assembly (with Fixed Angle Assembly)	0.83	9.29×10^{-4}
Axle Assembly (no Endcap)	0.18	1.31×10^{-4}
Axle Assembly (with Endcap)	0.32	2.41×10^{-4}
Mounting Assembly (with 25mm Truncated Endcap)	0.33	2.04×10^{-4}
Mounting Assembly (with Offset Assembly and extra Endcap)	1.63	1.67×10^{-3}
Mounting Assembly (with 50mm Truncated Endcap)	0.36	2.25×10^{-4}

Table 5.1: Estimated mass and inertia values for the constituent parts and sub-assemblies making up the reconfigurable test rig

which a motor would have to exert on the mechanism crank to propel each mechanism at a constant velocity of 100 rpm throughout a complete cycle of crank. The results of this inverse dynamic analysis can be seen in Fig. 5.18.

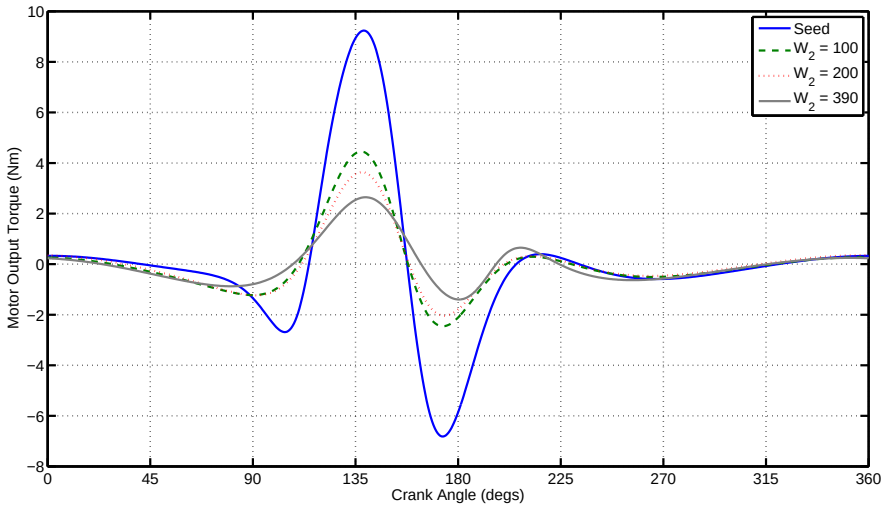


Figure 5.18: The estimated variation in demand torque need to propel the mechanism crank of a reconfigurable test rig configured in the form the seed mechanism and synthesised mechanisms at a constant velocity of 100 rpm over a complete cycle of crank

The variations in peak-to-peak magnitude of the resultant torque demands are summarised in Fig. 5.19.

Analysis of Figs. 5.18 and 5.19 show that, as predicted through the work discussed in Ch. 4, all of the alternative mechanism designs should be dynamically superior to the seed mechanism. Analysis of Fig. 5.19 shows that the dynamically most desirable mechanism design is that corresponding to $W_2 = 390$, which demonstrates a torque demand peak-to-peak magnitude reduction of 74.82% when compared to the same response of the seed mechanism. The mechanism corresponding to $W_2 = 200$ was the second most effective with a torque demand peak-to-peak magnitude reduction of 64.67% and the mechanism corresponding to $W_2 = 100$ being the third most effective with a torque demand peak-to-peak magnitude reduction of 56.96%. This result contrasts with the simulation work carried out in Ch. 4, in which the mechanism design corresponding to $W_2 = 100$, neglecting the design corresponding to $W_2 = 300$, was seen to be the most

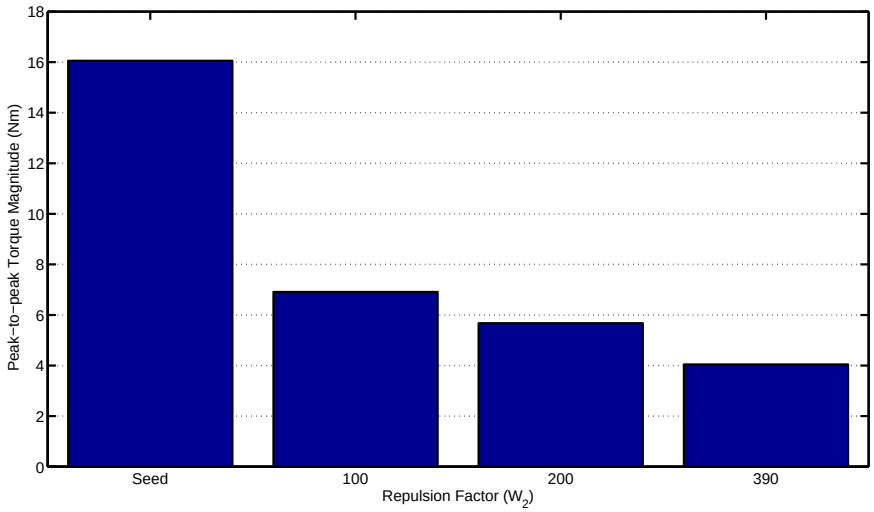


Figure 5.19: A comparison of the estimated torque demand peak-to-peak magnitudes required to propel the mechanism crank of a reconfigurable test rig configured in the form the seed mechanism and synthesised mechanisms at a constant velocity of 100 rpm over a complete cycle of crank

effective. The reason for this hierarchical change can be attributed to the construction of the parts of the constituent parts of the reconfigurable test rig differing from the construction of the parts assumed during the synthesis process. For example during the synthesis process, the centres of mass of all the links were considered to lie at their centre points. Also, the masses and intertias of the inter-link connections were considered to be small and therefore neglected. When modelling the reconfigurable test rig, it was not longer applicable to model the mechanism in this way. The masses and intertias of the inter-link connections of the reconfigurable test rig were seen to be considerable and so could not be neglected in the resultant dynamic models. Also the centre of masses of the links did not necessarily lie at the centre point of the links. These simulations show that these variations are theoretically sufficient to change performance hierarchy of the synthesised mechanisms, highlighting the importance of accurately and thoroughly defining the construction of constituent parts prior to performing mechanism synthesis.

5.3.1 System Parameter Estimation

As discussed in Ch. 2 and in accordance with the model of the test rig and control architecture, it was necessary to carry out experimental work to derive values for a series of physical system parameters, namely:

1. Inertia of motor and belt drive transmission and crank, J_{drive}
2. Coefficient of motor and belt drive transmission, b_{comb}
3. Combined controller gain of BRU-DDM30 and Deva004, K_m
4. The resultant gain of signal conditioning filters in the Deva004, K_{filter}

All parameter estimation work was carried out with the reconfigurable test rig assembled in the configuration of the seed mechanism. The mechanism crank was then disconnected from the outer housing of the driven ground point such that the motor only propelled the belt drive transmission and the outer housing of the driven ground point. With the crank removed, the M8 screws, link sleeves and 25 mm truncated endcap, which would ordinarily fasten the crank linkage to the housing section were reattached to the housing. In this way a single inertia value could be derived describing these components only. This inertia value varies in definition from that of J_{drive} since a crank in no longer included in the assembly. This new inertia value will therefore be referred to as $J_{driveNoCrank}$. In this configuration, the dynamics of the components being actuated theoretically approximate to those of a simple, linear, first order load simplifying the analysis process. The resultant apparatus set up is shown in Fig. 5.20.

The motion control scripts written to control the Woodpecker mechanism test rig described in Ch. 3, were reused to control this test rig. The test rig was configured such that only proportional control was used to govern both velocity and position system control loops, whereby $Ki_{vel} = 200$ and $Kp_{pos} = 100$. The system was actuated using an 810 rpm step velocity demand signal. It was actuated for a sufficiently long time,

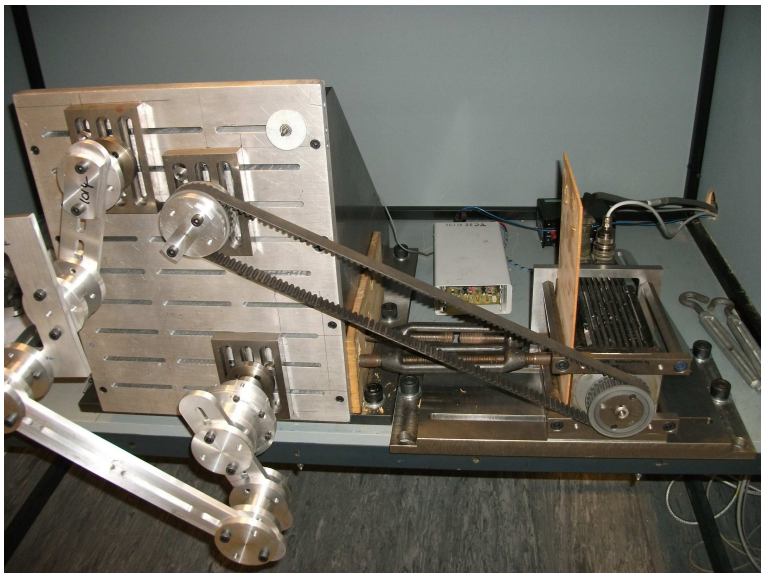


Figure 5.20: The reconfigurable test rig in its simplified configuration

to allow the rig to achieve steady state conditions. The response of the system to this signal was logged using the DAQ and the Deva004. The BRU-DDM30 is capable of monitoring and outputting only two analogue system status signals at any one time. These signals were logged using the DAQ. The BRU-DDM30 was configured such that it monitored and output, drive output current, I_{out} (used to calculate motor torque) and motor position, P_{out}.

The Deva004 was configured to monitor, demand and output motor velocities, V_{dem} and V_{out} respectively, as well as demand and output motor positions, P_{dem} and P_{out} respectively. The logging of P_{out} was duplicated by the Deva004 and by the DAQ. This duplicated data was used to align the data collected by the DAQ with that collected by the Deva004.

Values for $J_{driveNoCrank}$, b_{comb} and K_m for the system propelling the simple first order load were derived using the method described in Sec. 2.4.

Thus:

$K_m = 3.4374$

$b_{comb} = 0.007$

$J_{driveNoCrank} = 0.0031\ kgms^{-1}$

Using these parameters, the simulated torque and velocity responses were generated. These simulated responses were compared to the equivalent experimentally derived responses as shown in Fig.5.21.

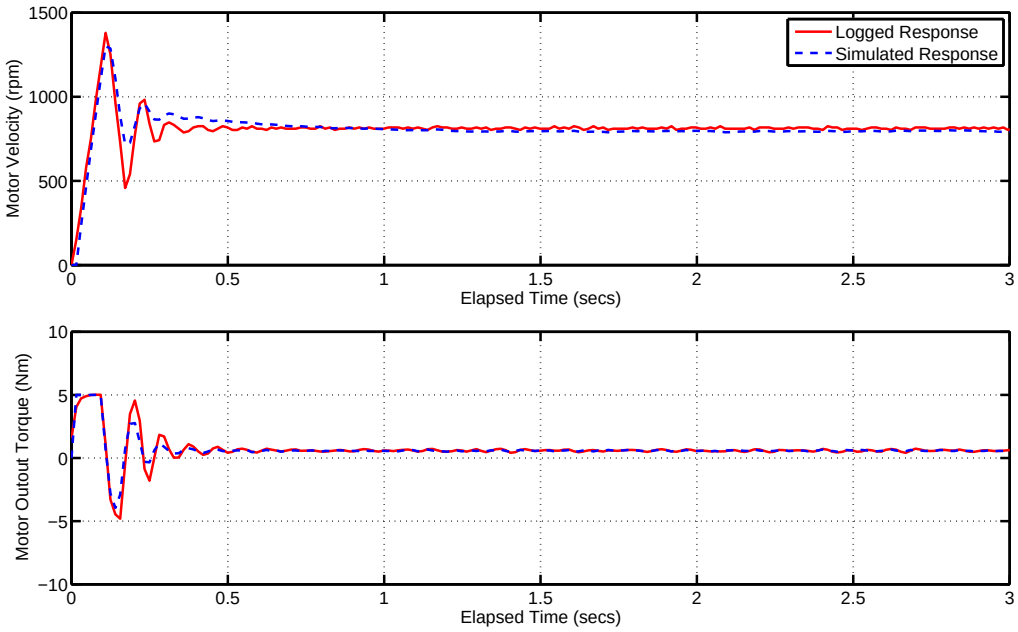


Figure 5.21: Comparing the simulated and experientally logged response of the reconfigurable test rig in a simplified configuration in response to a 810 rpm motor velocity step demand signal ($Kp_{vel} = 200$, $Ki_{vel} = 0$ and $Kp_{pos} = 100$)

The simulated and experimentally derived torque and velocity responses can be seen to correlate very closely in both transient and steady state portions of the response, indicating that the system in this configuration had been well modelled.

Work was carried out to model the complete test rig including, the mechanism. To achieve this, the mechanism crank was reconnected to the driven housing allowing the complete mechanism to be propelled by the motor. The DYSIM model of the mechanism was updated to take into account the newly derived inertia and friction coefficient values.

Initially, controller gains were set to $Kp_{vel} = 200$ and $Kp_{pos} = 100$, replicating the control parameters used in the test rig in Ch. 5. The motor was actuated using a 100

rpm velocity step demand signal. With a 1:1 gear ratio existing between the motor output shaft and the mechanism crank, this corresponds to a 100 rpm velocity demand signal being applied to the mechanism crank. The response of the system was logged.

Using the model of the reconfigurable test rig, a computer simulation was performed to estimate the response of the test rig to the velocity step demand signal. The simulated response was compared to the experimentally derived response. The simulated and experimentally derived responses did not correlate well however, with significant discrepancies in the steady state portion of the motion indicating an incorrect value of the system coefficient of friction, b_{comb} being used. As detailed in Sec. 2.2, work was done to increase the coefficient of friction value, b_{comb}, to take into account additional friction induced by the bearing units in inter-link connections. The same situation existed in this experiment and so the discrepancy was not unexpected. Further simulations were run, using a wide variety of coefficient of friction values. Each response was compared, in turn to the experimentally derived response. Unfortunately no value of b_{comb} could be identified which provided a satisfactory level of fit between the experimental and simulated data. A summary of these simulation attempts can be seen in Fig. 5.22.

At low coefficient of friction values, it was possible to simulate the magnitudes of the velocity peaks but the simulated magnitudes of the velocity troughs were too small. Increasing the b_{comb} allowed for larger magnitude velocity troughs but the generated velocity peaks were too small. The converse behaviour was true for torque responses.

Further analysis of the experimental velocity and torque responses of the reconfigurable test rig yielded a possible explanation as to why the modelling the test rig was proving difficult. It can be seen that as the mechanism passes through the region of non-linearity, where the velocity of the mechanism decreases, the velocity of the mechanism crank falls to a very low value, far below the demanded 100 rpm. Elsewhere in the response, the mechanism crank is moving at a higher velocity.

Figure 5.23 [1] depicts the variation in the coefficient of friction that persists when two lubricated metal plates move whilst in contact with each other. Line I shows the

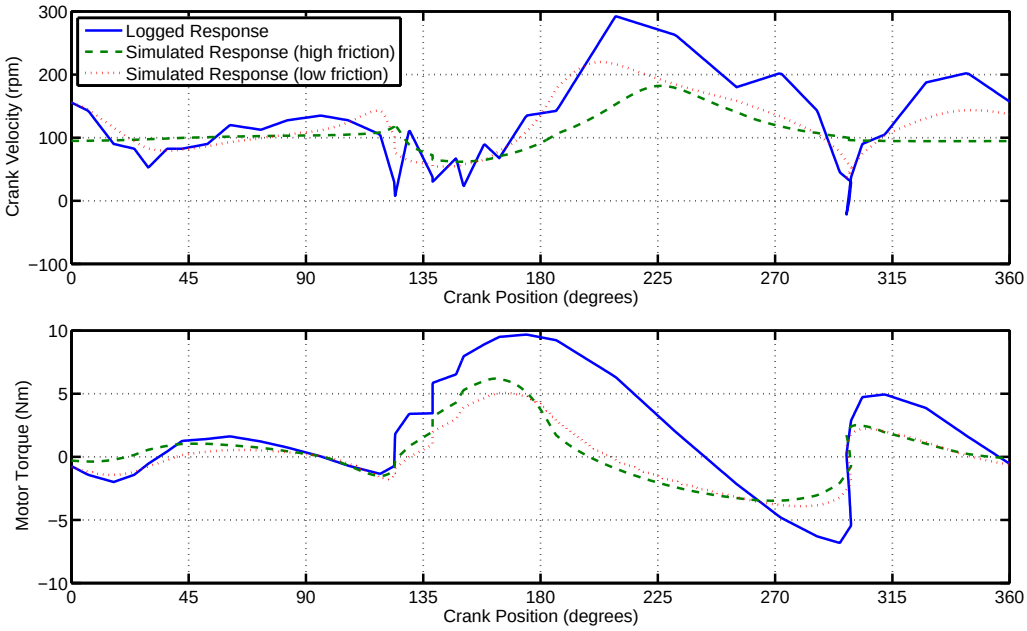


Figure 5.22: Comparing the simulated and experimentally logged responses of the reconfigurable test rig configured in the form of the seed mechanism (Woodpecker mechanism) in response to a 100 rpm crank velocity step demand signal using high and low coefficient of friction (b_{comb}) values ($Kp_{vel} = 200$, $Ki_{vel} = 0$ and $Kp_{pos} = 100$)

variation in value of the coefficient of friction, μ_s, that persists during the static friction behavioural region, which occurs when two contacting surfaces accelerate from rest to instantaneously assume a target speed. By contrasting line II depicts the variation in the coefficient of friction value, μ_k, that persists during the dynamic friction behavioural region, which occurs when two contacting surfaces move at a constant speed over each other. Line III shows the variation in coefficient of friction, $\Delta\mu$, which persists when two contacting surface move with an oscillatory motion from rest to a target speed.

Of particular interest to the motion under scrutiny are lines I and II. Consider a body on a surface that is initially at rest. In response to step velocity demand signal the body will initially accelerate rapidly in response to the rising edge of the signal, to a target speed, before assuming steady state conditions. When accelerating from rest, the motion of the body will be governed by static coefficient of friction regime, with μ_s governing the motion. As the body assumes steady state conditions, the static coefficient of friction regime will end and the dynamic coefficient of friction regime will start. μ_k will then govern the motion of the body. Once in motion, velocity fluctuations

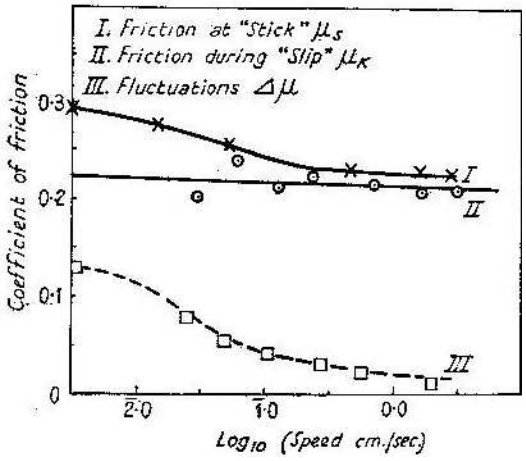


Figure 5.23: Typical variation of coefficient of friction with operating speed [1]

will take place under the dynamic friction behavioural regime, so long as the body does not come to rest at any time. If the body does come to rest, then the dynamic friction regime will end and the static friction regime will start again.

In all previous experimental system responses discussed so far in this thesis, with the exception of the response depicted Fig. 5.22, no steady state velocity responses have been encountered in which the mechanism crank has decelerated to rest. Instead, the mechanism crank has always been rotating at considerable speed. In all models considered so far, friction has been modelled in terms of a single coefficient value, b_{comb} derived using the steady state velocity response of the system. This coefficient value was therefore derived whilst the system is operating in the dynamic friction regime and thus this parameter is equivalent to μ_k. Analysis of Fig. 5.23 shows that the μ_k barely varies at all with actuation speed, confirming the validity of the modelling approach for the cases being considered.

In contrast, in the experimental steady state response depicted in Fig. 5.22 a steady state response is experienced where the mechanism crank does come to rest. In this case, the system moves from the dynamic friction regime to the static friction regime, causing the coefficient of friction governing the motion of the system to change. This phenomenon has not been modelled, thus accounting for the difficulty in matching the experimental and simulated responses, as depicted in Fig. 5.22.

To overcome this problem work was done to improve the performance of the test rig with the aim of preventing the mechanism crank from coming to rest and moving out of the dynamic friction operating region during steady state operation, resulting in a smoother, faster velocity response. This was done by tuning the velocity controller gains. The following gain combination was identified to produce a satisfactory velocity response:

$Kp_{vel} = 1300$
$Ki_{vel} = 500$
$Kp_{pos} = 100$

Using these new controller gain values, the steady velocity performance of the system was much improved, with a much smoother velocity response. The model of the system was updated to reflect the new controller gain values and computer simulations run. The simulated responses of the test rig were again compared to the experimentally derived responses. A coefficient of friction for the system was identified, which generated a good level of fit between the simulated and experimentally derived responses. Whereby:

$b_{comb} = 0.1$

Using these parameters, it was possible to match accurately the magnitude of the torque and velocity peaks and troughs as shown in Fig. 5.24.

The phasing and magnitude of the simulated and experimental peaks and troughs were also shown to align in terms of crank position, indicating that the dynamics of the mechanism had been well modelled. With such good correlation between the experimental and simulated velocity and torque steady state responses, the model of the system was determined to be accurate and fit for further work.

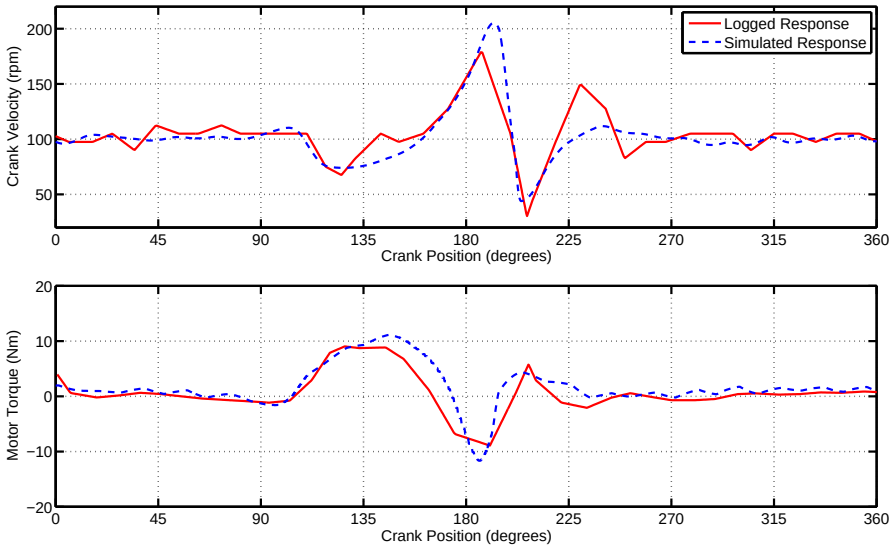


Figure 5.24: Comparing the simulated and experimentally logged torque and velocity response of the reconfigurable test rig configured in the form of the seed mechanism (Woodpecker mechanism) in response to a 100 rpm crank velocity step demand signal ($Kp_{vel} = 1300$, $Ki_{vel} = 500$ and $Kp_{pos} = 100$)

5.4 Conclusions

In this chapter work was discussed, in which the a new reconfigurable laboratory test rig was designed and constructed. The reconfigurable test rig had been designed to be versatile and possess the capability to be easily assembled in a wide variety of configurations. Using this test rig, it was possible to assemble mechanisms with varying geometries and styles without the need for the refabrication of parts. The propulsion and data logging elements of the test rig were adapted from the Woodpecker mechanism test rig discussed in Ch. 5.

With the reconfigurable test rig set up to mimic the original Woodpecker mechanism, work was performed to model the system. The resultant model was seen to accurately reflect the steady state behaviour of the real test rig.

Inverse dynamic analysis was performed to simulate the dynamics of the reconfigurable test rig, configured in the geometries of the mechanisms synthesised in Ch. 4. The mechanisms were simulated cycling at 100 rpm. The seed mechanism was also simulated.

The simulation showed that all the alternative mechanism designs should be superior to the seed mechanism, with the most superior mechanism design, corresponding to $W_2 = 390$, being predicted to the most superior. This mechanism design demonstrated a peak-to-peak torque demand magnitude reduction of 74.82%, compared to the equivalent torque demand of the seed mechanism. The hierarchy of effectiveness of the mechanisms modelled in the form of the reconfigurable test rig was shown to vary from the hierarchy identified in Ch. 4. This variation was attributed to changes in mechanism construction with the inclusion of significant masses and inertias at the rolling connection points between links and the locating of centres of mass away from the central point of the links, in the case of the reconfigurable test rig. This result demonstrated the importance of accurately and thoroughly defining the construction of the constituent mechanism parts prior to performing mechanism synthesis.

Chapter 6

Experimental Work Using The Reconfigurable Test Rig

In Ch. 5, the reconfigurable mechanism test rig was designed, assembled and modelled. In this chapter work will be discussed in which this test rig is used to experimentally verify the simulation work carried out in Ch. 4, in which a series of mechanism designs were synthesised. Each synthesised mechanism design was modelled and simulations were carried out, to estimate the dynamic performance of each design. The simulations showed each design to possess varying levels of dynamic performance. Using the test rig, these mechanism designs will be assembled and analysed for dynamic quality.

Work will also be discussed in which the Cam Function Generation Method, discussed in Ch. 3, will be applied to selected mechanism designs. The method will be applied to the test rig assembled in two configurations. The method will first be applied to the test rig assembled in the configuration of the seed mechanism and secondly in the configuration of the most dynamically superior mechanism. Applying the Cam Function Generation Method to the most dynamically superior mechanism design will be highly significant, as it will represent the combination of the two analytical methods discussed in this project. Analysis of the resultant response will demonstrate the effects

of combining the two methods.

6.1 Spectral Analysis

As implied in the literature [27,31] and demonstrated by simulation work, reducing the peak-to-peak magnitude of the variation in torque that the drive motor must exert on a system to generate a given motion will result in a reduction in the amount of harmonic content in the output motion of the mechanism. In an attempt to measure the magnitude of the harmonic content induced by the motion of the test rig, an accelerometer was mounted to the baseplate, as discussed in Sec. 5.1.2, to detect vibrations in the vertical plane. Measurements from the accelerometer were logged using the DAQ.

The logged data from the accelerometer was processed using Fourier analysis [62]. Using this method, the vibrational data was broken down into its constituent frequencies, allowing the makeup of the harmonic content of the test rig to be seen. Due to the significant stiffness of the mechanism base plate it was deemed acceptable to assume that vibrations experienced at the baseplate were also directly experienced by the mechanism during actuation. Due to the rigid connection between the baseplate and the supporting table surface, the same vibrational forces also transmitted to the other elements of the test rig and vice versa. It was observed that the table was vibrating when the mechanism was actuated. The logged vibrations detected by the accelerometer can therefore be seen to comprise not only the harmonic content of the mechanism but also the induced vibration of the rest of the supporting structure of the test rig.

In an attempt to distinguish between vibrations induced by the table and supporting structure and those induced by the mechanism, a simple *hammer* test was performed. A command script was run to instruct the motor to hold the mechanism in a fixed, stationary position. The test rig was assembled in the configuration of the seed mechanism. Using a soft hammer, the bottom of the table was struck sharply, inducing a pulse, shock input to the system. The resultant vibrational response was logged using

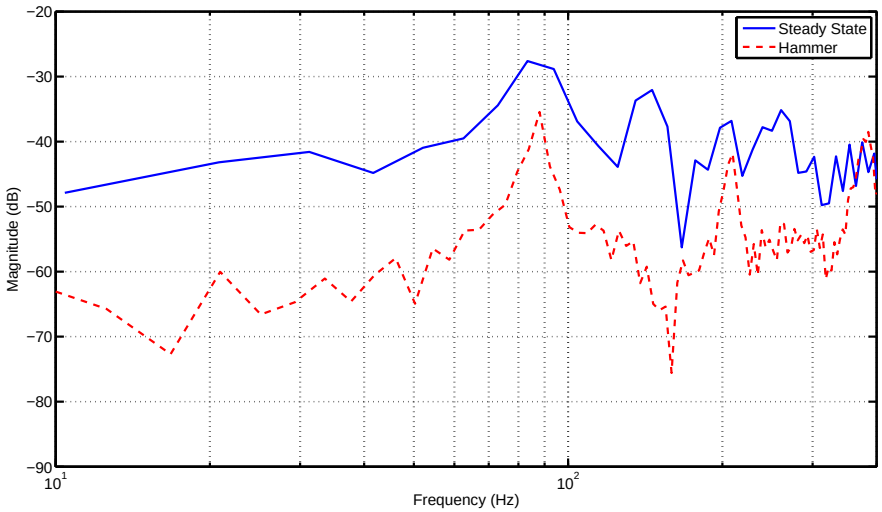


Figure 6.1: Comparing the frequency responses of the reconfigurable test rig configured in the form of the seed mechanism (Woodpecker mechanism) during steady actuation in response to a 100 rpm crank velocity step signal and whilst static in response to a pulse (hammer) signal ($Kp_{vel} = 1300$, $Ki_{vel} = 500$ and $Kp_{pos} = 100$)

the accelerometer and the data analysed using a Fourier analysis. With the mechanism stationary and the effectively locked in position by the motor, the dominant resonant frequencies should be those of the table and supporting structure. The resultant frequency response was compared with the corresponding frequency response induced by the mechanism operating under steady state conditions at 100 rpm. These responses are shown in Fig. 6.1.

Analysis and comparison of the two responses, shown in Fig. 6.1, revealed two highly similar frequency responses. The magnitudes of the two responses differ, as differing actuating forces were used to excite the system. Despite this, a close correlation between the positioning of the resonant peaks present in both responses can be seen. Using this result however, it was not possible to conclusively distinguish between the harmonic content induced by the mechanism during motion and the resonant vibration characteristics of the supporting structure. This result suggests that the frequency response of the mechanism is being masked by the frequency response of the table and supporting structure thus making it difficult to distinguish between the two responses and directly quantify the induced vibrations. The use of a stiffer and more rigid supporting

structure would greatly diminish this masking phenomenon.

6.2 Comparing the Alternative Mechanism Designs

The primary function of the reconfigurable test rig, discussed in Ch. 5, was to provide an easy method of assembling and analysing the performance of different mechanism designs without the need for the refabrication of parts. With the construction of such a test rig, it was easily possible to assemble the seed mechanism and all but one of the selected alternative mechanism designs, discussed in Ch. 4, and to experimentally analyse their dynamic behaviour.

In Ch. 5, the test rig was configured in the form of the seed mechanism and actuated using a 100 rpm velocity step demand signal. The system was excited for long enough to allow steady state conditions to be achieved. The dynamic behaviour of the system was then analysed. Each of the three remaining alternative mechanism designs, namely those corresponding to $W_2 = 100$, $W_2 = 200$ and $W_2 = 390$ were assembled in order of increasing value of W_2 and actuated using a 100 rpm velocity step demand signal. The response of the systems in each configuration were logged and analysed.

6.2.1 Results

Once the dynamic responses of all four mechanism configurations to a 100 rpm velocity step demand signal had been recorded, the steady state portion of each response was identified. The velocity and torque responses of the test rig in each mechanism configuration were then compared and analysed. A summary of the steady state torque and velocity responses of the four mechanism designs, can be seen in Fig. 6.2.

The experimentally derived results, shown in Fig. 6.2, correlate with the simulation results discussed in Ch. 5 obtained using inverse dynamics. The simulation work predicted that the alternative mechanism designs would demonstrate steady state torque

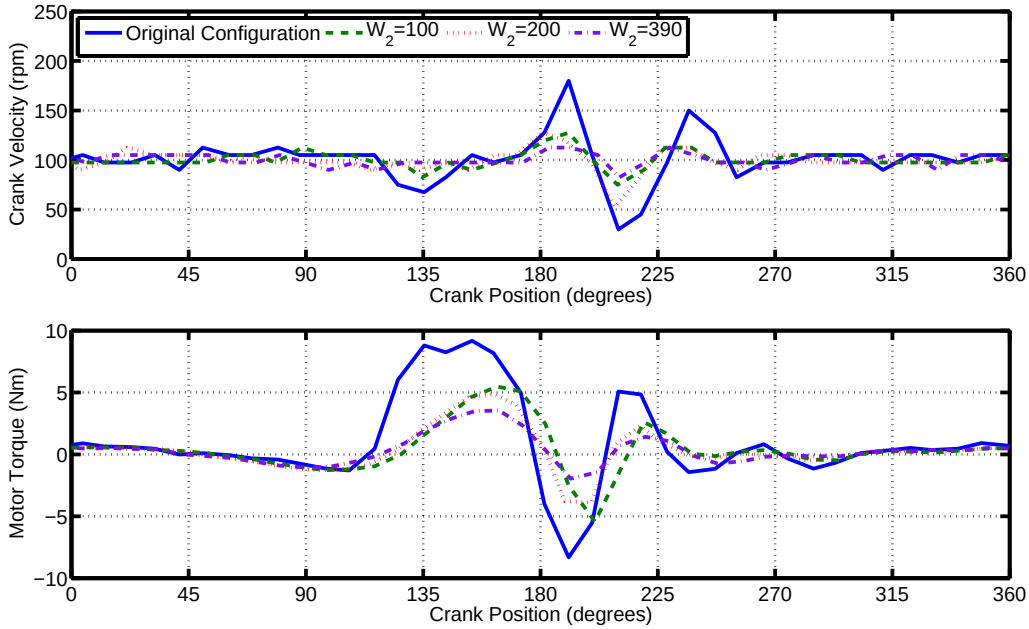


Figure 6.2: Comparing the experimentally logged steady state velocity torque responses of the reconfigurable test rig configured in the form of the seed mechanism (Woodpecker mechanism) and the selected synthesised mechanisms in response to a 100 rpm crank velocity demand signal ($Kp_{vel} = 1300$, $Ki_{vel} = 500$ and $Kp_{pos} = 100$)

variations with smaller peak-to-peak magnitudes than the seed mechanism as shown in Fig. 4.8. This prediction was borne out in the experimental work. It was also predicted that the peak-to-peak magnitude of the torque variations would decrease as the parameter W_2 is increased. This prediction was also confirmed by the experimental results. Analysis of the experimental results, shown in Fig. 6.2, show that the most superior mechanism design corresponds to $W_2 = 390$. In response to a 100 rpm velocity step demand signal this mechanism design required a steady state demand torque variation with a peak-to-peak magnitude of 5.54 Nm. To generate the same motion, the seed mechanism required a demand torque variation with a peak-to-peak magnitude of 17.49 Nm. Thus the alternative mechanism demonstrated a reduction in torque demand peak-to-peak magnitude of 68.3%.

This experimental work confirmed the simulation results, which predicted that significant dynamic performance benefits could be achieved through the resynthesis of the seed mechanism, using a method that takes into account both mechanism dynamics and kinematics.

6.3 Applying The Cam Function Shaping Method To Test Rig

In Ch. 3, a method entitled the Cam Function Generation Method was developed. Using this method, it is possible to derive motion and system specific variable velocity demand signals with which can improve the dynamic performance of a test rig. Following extensive simulation work, the method was successfully applied to the Woodpecker mechanism test rig, as discussed in Ch. 3. To reiterate the effectiveness of the method, work was carried out to apply the Cam Function Generation method to the new reconfigurable test rig. The method was applied to the test rig in two mechanism configurations, namely that of seed mechanism and that of the most superior alternative configuration ($W_2 = 390$). In each case, a target steady state cyclic rate of 100 rpm was sought.

Applying the Cam Function Generation Method to the reconfigurable test rig in its seed mechanism configuration allowed the effect of the method on the test rig to be compared with the impact of the method on the dynamics of the Woodpecker mechanism test rig. Applying the Cam Function Generation Method to the most superior alternative mechanism however, represents the combination of the Cam Function Generation Method and the mechanism synthesis strategy discussed in Ch. 4 allowing the potential benefits of combining both methods to be demonstrated.

6.3.1 Results

Figure 6.3 compares the steady-state velocity responses and torque demands of the test rig to a synthesised cam function generating a 100 rpm cyclic rate, with the equivalent responses to a 100 rpm velocity step demand signal. The test rig was assembled in the form of the seed mechanism.

Analysis of Fig. 6.3 shows that by using a cam function the magnitude of the peak-

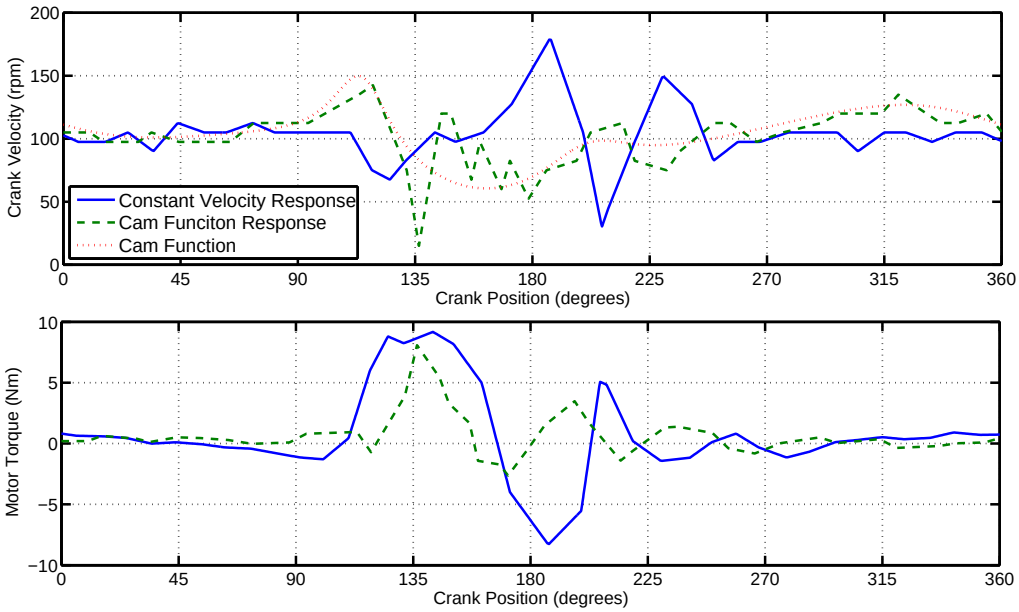


Figure 6.3: A comparison of the experimentally derived steady state velocity and torque responses of the reconfigurable test rig configured in the form of the seed mechanism (Woodpecker mechanism) in response to a 100 rpm crank velocity step demand signal compared with a synthesised cam function for operation at an average crank cyclic velocity of 100 rpm ($Kp_{vel} = 1300$, $Ki_{vel} = 500$ and $Kp_{pos} = 100$)

to-peak torque demand of the system was reduced from 17.49 Nm to 10.709 Nm, a reduction of 38.8%. If this variation in torque demand is compared to the equivalent torque demand variation of the Woodpecker mechanism test rig, as depicted in Fig. 3.13 an interesting level of similarity between the two torque demand variations can be seen. It can be seen that in both cases, the region of the motion where the greatest reduction in peak-to-peak magnitude is experienced, is in the region of the motion where, in response to the velocity step demand signal, there would have been a large torque trough. Using the cam function, the magnitude of this torque trough is greatly reduced.

The cam function shaping method was also applied to the test rig when configured in the geometry of the mechanism corresponding to $W_2 = 390$. The mechanism configuration had been shown to be the dynamically most superior alternative design. A comparison between the velocity and torque responses of the system to a 100 rpm constant velocity demand signal was compared to the equivalent responses to a cam function are shown in Fig. 6.4.

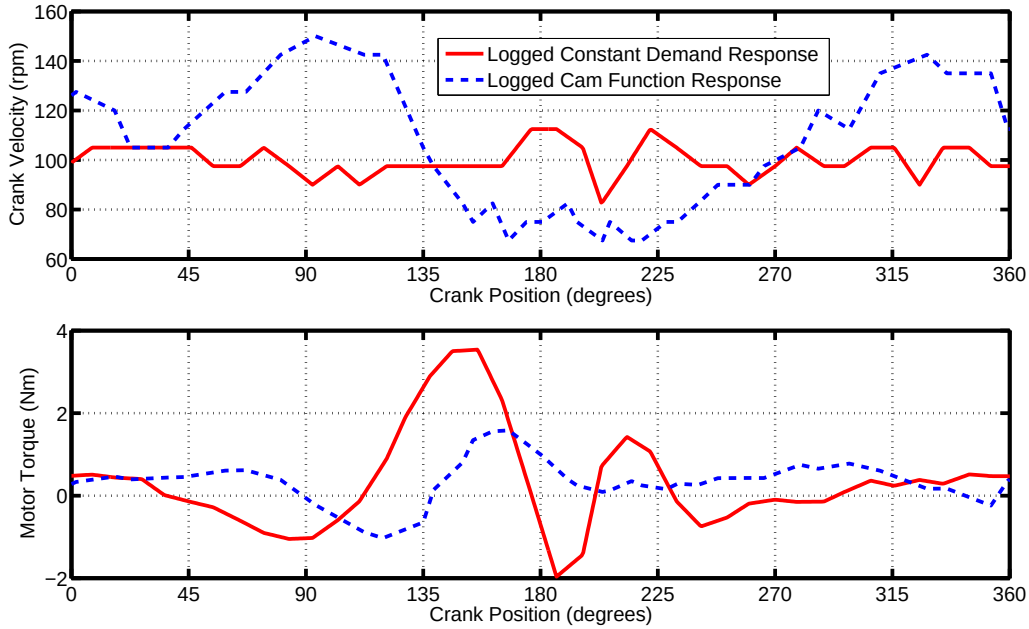


Figure 6.4: A comparison of the experimentally derived steady state torque and velocity responses of the reconfigurable test rig configured in the form of the dynamically most superior mechanism design ($W_2 = 390$) in response to a 100 rpm crank velocity step demand signal compared with a synthesised cam function for operation at an average cyclic crank velocity of 100 rpm ($Kp_{vel} = 1300$, $Ki_{vel} = 500$ and $Kp_{pos} = 100$)

Analysis of Fig. 6.4 shows that using a cam function generating a 100 rpm cyclic velocity in place of 100 rpm velocity step demand signal, the peak-to-peak magnitude of the torque demand signal was reduced from 5.54 Nm to 2.609 Nm, a reduction of 52.9%.

Figure 6.5 compares the torque and velocity responses of the reconfigurable test rigs in two different configurations and being actuated using two different velocity demand signals. The first set of responses depict the behaviour of the test rig in the configuration of the seed mechanism, being actuated by a 100 rpm constant velocity demand signal. The second set of response however, show the behaviour of the test rig in the configuration of the dynamically most superior alternative mechanism design, being actuated at a cyclic rate of 100 rpm by a cam function.

From Fig. 6.5, it can be seen that by resynthesing the seed mechanism and actuating the resultant alternative mechanism using a cam function instead of a constant velocity demand signal, the peak-to-peak magnitude of the torque demand was reduced from 17.49 Nm to 2.609 Nm, a reduction of 85.1%, a highly significant reduction. This

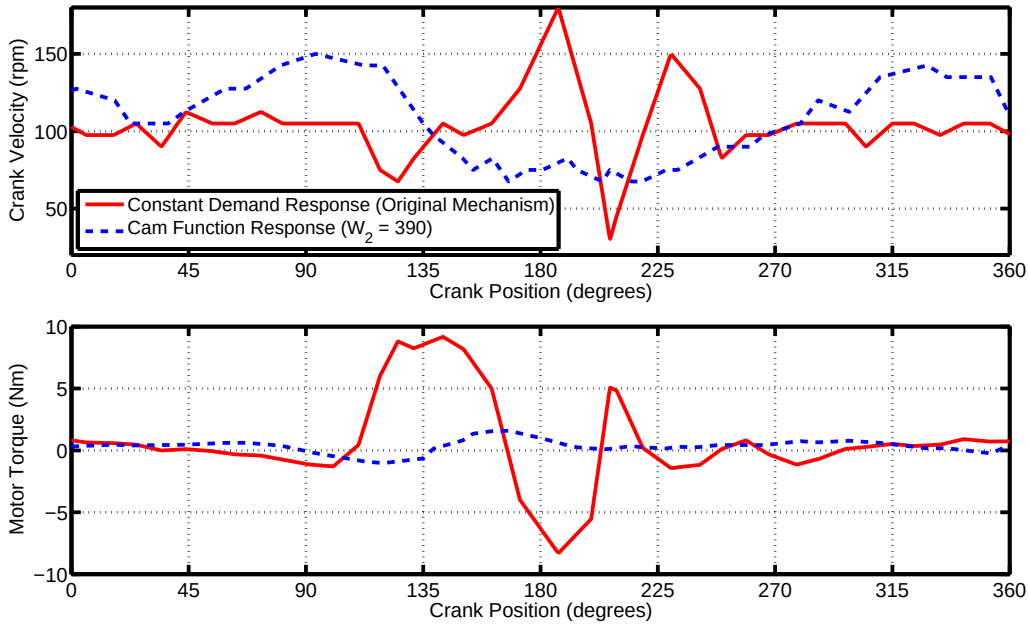


Figure 6.5: A comparison of the experimentally derived steady state torque and velocity responses of the reconfigurable test rig configured in the form of the seed mechanism (Woodpecker mechanism) and the configuration of the dynamically most superior mechanism design ($W_2 = 390$). The seed mechanism was actuated using a 100 rpm crank velocity step demand signal whilst the dynamically most superior mechanism was actuated using a 100 rpm cam function. ($Kp_{vel} = 1300$, $Ki_{vel} = 500$ and $Kp_{pos} = 100$)

clearly demonstrates the great benefits of synthesising a mechanism for both kinematic and dynamic performance and then propelling the mechanism using a variable velocity cam function generated using The Cam Function Generation Method.

Although it was not possible to quantify any reductions in harmonic content being induced by the motion of the mechanism, it was obvious that the most superior alternative mechanism was significantly quieter than the seed mechanism during operation when actuated using a 100 rpm constant velocity demand signal. The mechanisms were also noticeably quieter still when actuated using cam functions.

6.4 Conclusions

In this chapter work was carried out to verify experimentally, the simulation work discussed in Chapters 3 and 4.

The reconfigurable test rig, described in Ch. 5, was used to assemble representations of each of the alternative mechanism designs generated using the mechanism synthesis method described in Ch. 4. The test rig was also assembled in the configuration of the seed mechanism. The experimental results correlated well with the simulation results. The simulations had predicted that as the synthesis weighting factor W_2 was increased, more dynamically desirable mechanisms would be obtained. This phenomenon was borne out in the experimental work. The seed mechanism responded to a 100 rpm constant velocity demand signal, with a steady state torque demand signal with a peak-to-peak magnitude of 17.49 Nm. The most superior alternative mechanism responded to the same demand signal with a torque demand signal with a peak-to-peak magnitude of 5.54 Nm, a reduction of 68.3%.

The Cam Function Generation method was applied to the test rig, configured in the form of the seed mechanism and also in the form of the most superior alternative mechanism. In the case of the seed mechanism, actuating the mechanism at 100 rpm using a cam function in place of 100 rpm constant velocity demand signal reduced the peak-to-peak magnitude of the steady state torque demand from 17.49 Nm to 10.79 Nm, a reduction of 38.8%. Carrying out the equivalent test with the test rig configured in the form of the dynamically most superior mechanism, demonstrated a reduction in peak-to-peak torque demand magnitude from 5.54 Nm to 2.609 Nm, a reduction of 52.9%.

If the peak-to-peak torque demand variation of the dynamically most superior mechanism design, actuated using a cam function generating a 100 rpm cyclic rate, is compared to the original seed mechanism being actuated using a 100 rpm constant velocity demand signal, then it can be seen that the magnitude of steady state peak-to-peak torque demand falls from 17.49 Nm to 2.609 Nm, a reduction of 85.1%. This final result is particularly significant as it demonstrates the power of synthesising a mechanism for both dynamic and kinematic performance and then propelling the resultant mechanism using a cam function.

Due to limitations in the test rig design it was not possible to quantify explicitly the amount of harmonic content in the output motions of the mechanism. Despite this however, by observing the test rig during operation, it was possible to hear audible reductions in mechanical noise compared to the original mechanism. The most superior mechanism was observed to be especially quiet, particularly when propelled using a cam function.

Chapter 7

Conclusions

In this thesis, two methods of improving the high speed performance of servo-mechanical machinery were developed. The methods allow servo-machinery to operate at high speeds with reduced levels of harmonic content in their output motions. As a basis for the development work of these methods, a prototype mechanism - nicknamed the Woodpecker mechanism - was provided by the project sponsor, ITCM Ltd. This mechanism was identified as having a harmonically rich output motion. The Woodpecker mechanism was installed on a pre-existing test rig, which originally lacked a servo motor, motor drive unit and motion controller.

It is implied in the literature that a direct link exists between the amount of harmonic content in an output motion and the magnitude of the peak-to-peak variation in torque that a drive motor needs to exert on a mechanism crank to generate that motion [27, 31]. For this reason, these methods use cyclic peak-to-peak output motor torque variation magnitude as a metric for quantifying reductions in harmonic content. Clearly, the smaller the peak-to-peak magnitude of the torque demand signal, the smaller the amount of harmonic content present in the output motion. Subsequent simulation work performed over the course of this project supported this link.

Using the Woodpecker mechanism test rig, work was carried out to specify and install

a servo motor, motor drive and motion controller. The resultant system utilised an Electro-Craft S-4075 servomotor connected to a BRU-DDM30 motor drive unit (operating as a torque amplifier), which accepted motion control signals from a Deva004 control card installed in a dedicated control computer. The mechanism crank was connected to the drive shaft of the motor using a synchronous belt drive transmission system. A data acquisition strategy was developed to allow system performance data to be logged.

Using a physical data obtained from a CAD model of the Woodpecker mechanism, a dynamic numerical model of the mechanism was created using a multi-body dynamic modelling tool entitled DYSIM. Using experimental data, obtained by exciting the test rig with a variety of velocity step and ramp demand signals, the remaining elements of the test rig were modelled. The experimental and simulated responses of the system to a velocity step demand signal were compared. With good levels of fit between the two responses, the model was deemed to be accurate.

Using this model, a method entitled the Cam Function Generation Method was developed. Using this method, it is possible to derive mechanism and motion specific, position dependent variable velocity demand signals, also known as *cam functions*, which allow a mechanism to operate at a target speed with reduced harmonic content in it output motion. The method was initially tested using computer simulations, with the application of the method to the Woodpecker mechanism test rig being simulated. Cam functions were developed that would propel the mechanism at average cyclic rates of 200, 300, 450 and 600 rpm. These responses were compared to the simulated responses of the test being actuated using equivalent constant velocity demand signals. Operating using 200 and 300 rpm cam functions, torque variation peak-to-peak magnitude reductions of 74.35% and 36.72% respectively, were experienced. Using a 600 rpm cam function, a reduction of 3.19% was simulated. Operating using a 450 rpm cam function, a peak-to-peak torque variation magnitude increase of 18.94% was simulated. It was hypothesised that the reason the method yielded less significant benefits, as the target speed deviated from 200 rpm, was due to the fact that the system parameters

used in the model were derived from experimental system responses, with the test rig being actuated at 200 rpm. It is therefore possible that these parameters do not accurately reflect the behaviour of the test rig being actuated at different speeds. Evidence supporting this hypothesis was provided through experimental work.

Spectral analysis was performed to compare the harmonic content in the output motion of the mechanism when actuated using a 200 rpm constant velocity demand signal to the corresponding harmonic content when the system is actuated using 200 rpm cam function. The analysis showed that the amplitude of the harmonic content in the output motion of the mechanism was smaller when using the cam function, particularly at higher frequencies. This result not only confirmed the effectiveness of the Cam Function Generation Method but also reiterated the link between peak-to-peak torque magnitude and harmonic content amplitude.

To experimentally verify the effectiveness of the Cam Function Generation Method, the Woodpecker test rig was actuated using a 200 rpm cam function and the steady state response of the system was compared with the equivalent system response when actuated with a 200 rpm constant velocity demand signal. Using the cam function the magnitude of the peak-to-peak torque variation fell from 5.6 Nm to 2.1 Nm, a reduction of 63%, thus confirming the effectiveness of the method.

The second method to be developed was a novel mechanism synthesis strategy, which synthesised mechanisms, taking into account both kinematics and dynamics. Using this method, starting with a original mechanism layout referred to as the seed mechanism, as well as a desired output path, alternative mechanism designs were synthesised which possess satisfactory kinematic behaviour as well as superior dynamic characteristics. A weighting parameter, W_2, was used to influence the synthesis process to generate different mechanism designs. The configuration and dimensions of the original Woodpecker mechanism was used as the seed mechanism. The synthesis process generated a series of alternative mechanism designs, four of which were selected for further analysis. Simulation work estimated that the most superior design would be the design corresponding

to a $W_2 = 300$. This design, if actuated using a 600 rpm constant velocity demand signal would require motor demand torque variation with a peak-to-peak magnitude 70% smaller than the equivalent demand for the seed mechanism. The three other selected mechanism designs showed lesser reductions in peak-to-peak torque variation magnitude. Spectral analysis showed that, when actuated using a 600 rpm constant velocity demand signal, the most superior alternative mechanism had an output motion with lower amplitude harmonic content than the equivalent motion carried out by the seed mechanism.

For the next stage of the work, a new reconfigurable test rig was designed and constructed. This test rig possessed the specific ability to be assembled in a variety of different geometries and configurations without the need for the refabrication of parts. This test rig reused the servo motor, motor drive, motion controller and data acquisition strategy used with the Woodpecker test rig. CAD models were created, simulating the assembly of the reconfigurable test rig in the form of the seed mechanism and the four alternative mechanism designs. This exercise showed that it was not possible to assemble the test rig in the form of the mechanism corresponding to $W_2 = 300$. This mechanism design was discarded. Using physical data taken from these CAD models, dynamic models of the test rig in these five mechanism configurations were constructed using DYSIM. Using these DYSIM models, inverse dynamic analysis was performed. This analysis showed that the alternative mechanism designs should be dynamically superior to the seed mechanism. In contrast to prior simulation work however, the inverse dynamic analysis showed a change performance hierarchy with the mechanism corresponding to $W_2 = 390$ now being the dynamically most superior mechanism. This change in performance hierarchy was attributed to changes in mechanism construction. Simulation work carried out during the synthesis stage of this work had assumed the masses and inertias of the inter-link joints to be negligible when compared with the masses and inertias of the links. All the links were also assumed to have their centres of mass located at their mid-points. In the case of the reconfigurable test rig however, the mass and inertia values of the inter-link joints were considerable and so could not be neglected. The centres of mass of the links were also not at the mid-point of the links.

These dynamic effects of these variations were sufficient to change the performance hierarchy. This result indicates the importance of accurately and thoroughly defining the construction of the constituent mechanism parts prior to performing mechanism synthesis.

Using the reconfigurable test rig, the seed mechanism, as well as the three remaining alternative mechanism designs were assembled and tested for dynamic performance using a 100 rpm constant velocity demand signal. The resultant performance hierarchy correlated well with hierarchy agreed with the hierarchy generated through the inverse dynamic simulation work. As predicted the mechanism design corresponding to $W_2 = 390$ was the most effective and the seed mechanism was the least effective. The most superior mechanism design, in response to a 100 rpm constant velocity demand signal responded with a motor torque variation with a peak-to-peak magnitude of 5.54 Nm. This response is 68.3% smaller than the 17.49 Nm peak-to-peak torque magnitude response of the seed mechanism.

The Cam Function Generation Method was then applied to the test rig. Cam functions were generated to actuate the seed mechanism and the dynamically most superior mechanism at a cyclic rate of 100 rpm. Experimental work showed that when applied to the seed mechanism design, use of the cam function resulted in motor torque variation with a peak-to-peak magnitude of 10.709 Nm. This is a reduction of 38.8% compared to the 17.49 Nm variation magnitude of the constant velocity demand signal.

When the dynamically most superior mechanism was actuated using a 100 rpm constant velocity demand signal, a peak-to-peak torque variation magnitude of 5.54 Nm was experienced. When the same mechanism was actuated using a 100 rpm cam function, a peak-to-peak torque variation magnitude of 2.609 Nm was experienced, a reduction of 52.9%. If this result is compared with the torque response of the seed mechanism being actuated using a 100 rpm constant velocity demand signal, this response represents a reduction of 85.1%. This result is particularly significant as it demonstrates the great benefits of synthesising a mechanism for both dynamic and kinematic quality and then

propelling it using a cam function.

An attempt was made to measure the amount of harmonic content in the output motions of the reconfigurable test rig. Due to the lack of stiffness in the supporting structure of the test rig it was not possible to achieve this. It was however, possible to observe, that the test rig in the configuration of the dynamically most superior mechanism design was significantly quieter during actuation than the test rig in the configuration of the seed mechanism. Further reductions in noise were observed when cam functions were used in place of constant velocity demand signals.

In this thesis two methods have been discussed and both have been proven to improve the dynamic performance of servo-machinery operating at high speeds. The two methods complement each other well as they tackle the problem from different angles. The Cam Function Generation Method is ideal for application to existing servo mechanisms, as the method does not require any physical modifications to the system to be made. The method merely requires the re-specification of a velocity demand signal. This method is therefore cheap and easy to implement. In contrast, the novel mechanism synthesis strategy deals with the problem of synthesising new mechanisms for high speed operation. Resultant mechanisms possess good dynamic and kinematic performance characteristics. To supplement this method, the Cam Function Generation Method can be applied to the resultant mechanism to fully exploit these characteristics and obtain an even higher quality dynamic system response.

Recommendations for further work to enhance and develop the methods have been suggested. These recommendations will not only improve the effectiveness of the methods, but will also increase their level of automation, making them more user friendly.

Bibliography

[1] F. P. Bowden and D Tabor. *The Friction and Lubrication of Solids, Part 1.* Oxford University Press, London, UK, 5 edition, 1971.

[2] J. M. Jimenez, G. Alvarez, J. Cardenal, and J. Cuadrado. A simple and general method for kinematic synthesis of spatial mechanisms. *Mechanism and Machine Theory*, 32(3):323–341, 1997.

[3] E. Fichter, G. Smith, P. Todd, and F. Wagner. *Mini-Atlas of Linkage Design.* Saltire Software Inc, 1994.

[4] A. Kirecci and L. C. Dulger. Motion design and implementation for a hybrid drive system. *Proceedings of the Institution of Mechanical Engineers Part I-Journal of Systems and Control Engineering*, 217(I4):299–302, 2003.

[5] A. Kirecci and L. C. Dulger. A study on a hybrid actuator. *Mechanism and Machine Theory*, 35(8):1141–1149, 2000.

[6] Z. Yuan, M. J. Gilmartin, and S. S. Douglas. Design of hybrid machines for nonuniform motion production. *Proceedings of the Institution of Mechanical Engineers Part C-Journal of Mechanical Engineering Science*, 219(5):491–499, 2005.

[7] M Stanisic. Mechanisms and machines. Electronic Citation, 2005.

[8] A. M. Connor, S. S. Douglas, and M. J. Gilmartin. The use of harmonic information in the optimal synthesis of mechanisms. *Journal of Engineering Design*, 9(3):239–249, 1998.

[9] Y Jin, I Chen, and G Yang. Mobility and singularity analysis of a selectively actuated parallel mechanism. 2004.

[10] Q. Jin and T. L. Yang. Theory for topology synthesis of parallel manipulators and its application to three-dimension-translation parallel manipulators. *Journal of Mechanical Design*, 126(4):625–639, 2004.

[11] R. Aviles, J. Vallejo, G. Ajuria, and J. Agirrebeitia. Second-order methods for the optimum synthesis of multibody systems. *Structural and Multidisciplinary Optimization*, 19(3):192–203, 2000.

[12] J. A. Cabrera, A. Simon, and M. Prado. Optimal synthesis of mechanisms with genetic algorithms. *Mechanism and Machine Theory*, 37(10):1165–1177, 2002.

[13] J. M. Hansen. Synthesis of mechanisms using time-varying dimensions. *Multibody System Dynamics*, 7(1):127–144, 2002.

[14] V. Jovanovic and K. Kazerounian. Optimal design using chaotic descent method. *Journal of Mechanical Design*, 122(3):265–270, 2000.

[15] M. A. Laribi, A. Mlika, L. Romdhane, and S. Zeghloul. A combined genetic algorithm-fuzzy logic method (ga-fl) in mechanisms synthesis. *Mechanism and Machine Theory*, 39(7):717–735, 2004.

[16] Y. Liu and R. B. Xiao. Optimal synthesis of mechanisms for path generation using refined numerical representation based model and ais based searching method. *Journal of Mechanical Design*, 127(4):688–691, 2005.

[17] X. C. Nie and V. Krovi. Fourier methods for kinematic synthesis of coupled serial chain mechanisms. *Journal of Mechanical Design*, 127(2):232–241, 2005.

[18] R. Sancibrian, F. Viadero, P. Garcia, and A. Fernandez. Gradient-based optimization of path synthesis problems in planar mechanisms. *Mechanism and Machine Theory*, 39(8):839–856, 2004.

[19] A. Saxena. Synthesis of compliant mechanisms for path generation using genetic algorithm. *Journal of Mechanical Design*, 127(4):745–752, 2005.

[20] A. A. Smaili, N. A. Diab, and N. A. Atallah. Optimum synthesis of mechanisms using tabu-gradient search algorithm. *Journal of Mechanical Design*, 127(5):917–923, 2005.

[21] Z. F. Yuan, M. J. Gilmartin, and S. S. Douglas. Optimal mechanism design for path generation and motions with reduced harmonic content. *Journal of Mechanical Design*, 126(1):191–196, 2004.

[22] L. P. Zhang, D. Wang, and J. S. Dai. Biological modeling and evolution based synthesis of metamorphic mechanisms. *Journal of Mechanical Design*, 130(7), 2008.

[23] K Price and R Storn. Differential evolution (de). Electronic Citation, 2006.

[24] A. A. Smaili and N. A. Diab. A new approach for exact/approximate point synthesis fo planar mechanisms. *Proceedings of IDETC/CIE*, pages 1–8, 2005.

[25] PS. Shiakolas, D. Koladiya, and J. Kebrle. On the optimum synthesis of six-bar linkages using differential evolution and the geometric centroid of precision positions technique. *Mechanism and Machine Theory*, 40(3):319–335, 2005.

[26] J. Mariappan and S. Krishnamurty. A generalized exact gradient method for mechanism synthesis. *Mechanism and Machine Theory*, 31(4):413–421, 1996.

[27] L. F. Yuan and J. S. Rastegar. Kinematics synthesis of linkage mechanisms with cam integrated joints for controlled harmonic content of the output motion. *Journal of Mechanical Design*, 126(1):135–142, 2004.

[28] L. P. Zhang, D. L. Wang, and J. S. Dai. Automated conceptual design for hybrid mechanisms based on characteristic state space theory. *Detc 2008: 32Nd Annual Mechanisms and Robotics Conference, Vol. 2, Pts A & B*, pages 697–706, 2009.

[29] R Stefani, B Shahian, C Savant, and G Hostetter. *Design of Feedback Control Systems*. Oxford University Press, New York, Oxford, 4th edition edition, 2002.

[30] E. R. Leal and J. S. Dai. From origami to a new class of centralized 3-dof parallel mechanisms. *Proceedings of the Asme International Design Engineering Technical*

Conferences and Computers and Information in Engineering Conference 2007, Vol 8, Pts A and B, pages 1183–1193, 2008.

[31] T. Rastegar and L. Yuan. A systematic method for kinematics synthesis of high-speed mechanisms with optimally integrated smart materials. *Journal of Mechanical Design*, 124(1):14–20, 2002.

[32] X. M. Wang. Electro-mechanical dynamic analysis of the piezoelectric stack. *Smart materials and structures*, 5(4):492–500, 1996.

[33] I. S. Kochev. General theory of complete shaking moment balancing of planar linkages: a critical review. *Mechanism and Machine Theory*, 35(11):1501–1514, 2000.

[34] T. W. Lee and C. Cheng. Optimum balancing of combined shaking force, shaking moment, and torque fluctuations in high-speed linkages. *Journal of Mechanisms Transmissions and Automation in Design-Transactions of the Asme*, 106(2):242–251, 1984.

[35] G. Alici and B. Shirinzadeh. Optimum dynamic balancing of planar parallel manipulators based on sensitivity analysis. *Mechanism and Machine Theory*, 41(12):1520–1532, 2006.

[36] F. L. Conte, G. R. George, R. W. Mayne, and J. P. Sadler. Optimum mechanism design combining kinematic and dynamic-force considerations. *Journal of Engineering for Industry-Transactions of the Asme*, 97(2):662–670, 1975.

[37] S. D. Yu and W. L. Cleghorn. Dynamic instability analysis of high-speed flexible four-bar mechanisms. *Mechanism and Machine Theory*, 37(11):1261–1285, 2002.

[38] W. L. Cleghorn, B. Tabarrok, and R. G. Fenton. Critical running speeds and stability of high-speed flexible mechanisms. *Mechanism and Machine Theory*, 19(3):307–317, 1984.

[39] W. L. Cleghorn, R. G. Fenton, and B. Tabarrok. Steady-state vibrational response of high-speed flexible mechanisms. *Mechanism and Machine Theory*, 19(4-5):417–423, 1984.

[40] Y. Q. Yu and M. R. Smith. The effect of cross-sectional parameters on the dynamics of elastic mechanisms. *Mechanism and Machine Theory*, 31(7):947–955, 1996.

[41] W. L. Cleghorn, R. G. Fenton, and B. Tabarrok. Optimum design of high-speed flexible mechanisms. *Mechanism and Machine Theory*, 16(4):399–406, 1981.

[42] S Antani, LR Long, GR Thoma, and DJ Lee. Shape representation and similarity algorithms. Report, 2006.

[43] R Alferez and YF Wang. Image indexing and retrieval using image-derived geometrically and illumination invariant features. In *International Conference on Multimedia Computing and Systems*, pages 177–182. IEEE Computer Society, Jun 1999.

[44] FHP Spaan, RL Lagendijk, and J Biemond. Shape coding usingpolar coordinate and the discrete cosine transform. *Proceedings of the 1997 International Conference on Image Processing (ICIP '97) 3-Volume Set-Volume 1 - Volume 1*, pages 516–519, 1997.

[45] G. Farin. *Curves and Surfaces for CAGD - A Practical Guide*. Morgan Kaufmann, San Diego, 5th edition, 2002.

[46] DF Rogers and NG Fog. Constrained b-spline curve and surface fitting. *Computer Aided Design*, 21(10):641–648, 1989.

[47] A Bowyer. Bernstein-basis curves and sculpted surfaces. Electronic Citation, 2004.

[48] MJD Powell. An efficient method for finding the minimum of a function of several variables without calculating derivatives. *The Computer Journal*, 7(2):155–162, 1964.

[49] Spatial Corp. Spatial:3d acis modeler. Internet Communication, 2007.

[50] Dare A. Wells. *Lagrangian Dynamics*. McGraw-Hill Inc. USA, 1967.

[51] The MathWorks Inc. Matlab 2009b. Computer Program, 9 A.D.

[52] M. N. Sahinkaya. Inverse dynamic analysis of multiphysics systems. *Proceedings of the Institution of Mechanical Engineers Part I-Journal of Systems and Control Engineering*, 218(I1):13–26, 2004.

[53] Reliance Motion Contol Inc. Bru master. Computer Program, 1994.

[54] Reliance Motion Contol Inc. *Electro Craft BRU-DDM30 Users Manual*. Reliance Motion Control, 1994.

[55] Deva Electronic Controls. Deva004 command library. Computer Program, 2006.

[56] M. N. Sahinkaya. Input shaping for vibration-free positioning of flexible systems. *Proceedings of the Institution of Mechanical Engineers Part I-Journal of Systems and Control Engineering*, 215(I5):467–481, 2001.

[57] Deva Electronic Controls. *DEVA004 Users Manual V3.0*. Deve Electronic Controls, 2007.

[58] K Dutton, S Thomptson, and B Barraclough. *The Art of Control Engineering*. Addison Wesley, Harlow, Essex, UK, 1997.

[59] Deva Electronic Controls. *DEVA004 SDK Programming Guide V1.3*. Deve Electronic Controls, 2007.

[60] M. N. Sahinkaya, RMC Rayner, RK Aggawal, G Vernon, and G Shirley. Synthesis of demand signals for high speed operation of a packaging mechanism. In *Proceedings of the ASME 2007 International Design Engineering Technical Conferences & Computers andInformation in Engineering Conference*, Sep 2009.

[61] JL Meriam and LG Kraige. *Engineering Mechanics: Dynamics*. John Wiley and Sons Inc, Toronto, Canada, 1998.

[62] Hwei P. Hsu. *Applied Fourier Analysis*. Harcourt Brace Jovanovich College Outline Series. Harcourt Brace Jovanovich Publisers, Orlando, Florida, USA, 1984.

[63] M Cianci. *Leonardo da Vinci's Machines*. Becocci Editore, Italy, Milan, 1988.

[64] PTC. Pro/engineer mechanism design solutions. Computer Program, 2005.

[65] UGS. Nx mechansim simulation. Computer Program, 2005.

[66] MSC Software Corporation. Mechanism/pro simulationof kinematic motion. Computer Program, 2005.

[67] BJ Hicks, AJ Medland, and G Mullineux. The representation and handling of constraints for the design, analysis and optimisation of high speed machinery. *Artificial Intelligence for Engineering Design, Analysis and Manufacture*, 20:313–328, 2006.

[68] PS Motion Ltd. Ps motion machine design software. Computer Program, 2008.

[69] SKF Group. Skf general catalogue. Catalog, 2007.

[70] Dexion Ltd. Dexion speedframe. Catalog, 2008.